
DICIEMBRE 1987
REVISTA PARA USUARIOS DE LA NORMA
LOAD
MSX
AÑO 2 Nº 20
A6,50 REP.ARGENTINA
FELIZ NAVIDAD
FREDDY HARDEST
4 SUPER 4
CAMELOT WARRIORS
OLE, TORO
GAME OVER
COMO COMANDAR TRENES Y AUTOS CON UNA COMPUTADORA
FACSIMILES EN MSX
SOFT INEDITO
MSX-DOS MISTERIOS DEVELADOS
DESNUDANDO LOS PROGRAMAS
GANADORES DEL CONCURSO

# A la computadora personal

# Talent MSX

# nada le es imposible

dialogo - *D.P.*




MSX es marca registrada de MICROSOFT CORPORATION

Porque gracias a la norma internacional MSX, la TALENT MSX trasciende sus propios límites.
Hasta ahora, cuando usted compraba una computadora personal de cualquier marca, quedaba automáticamente desconectado del resto del mundo de la computación. Porque los distintos equipos y sistemas no eran compatibles entre sí.
Hasta que dos grandes empresas de informática, la Microsoft Corp. de EE.UU. y la ASCII del Japón se pusieron de acuerdo para crear una norma standard: la MSX. Que se expandió también rápidamente en Europa. Y que hoy TALENT presenta por primera vez en la Argentina.
Mientras que la mayoría de las computadoras de su tipo que se ofrecen en el mercado nacional, han sido discontinuadas por obsoletas en sus lugares de origen, TALENT MSX tiene casi ilimitadas posibilidades de desarrollo. Porque la norma MSX es en todo el mundo inteligencia en crecimiento.
La TALENT MSX pone a su disposición un mundo de software para elegir. Y con la incorporación de todos sus periféricos llega a ser una auténtica computadora profesional.

## UTIL

Su poderoso sistema operativo MSX permite el acceso a todo tipo de procesamiento de datos:

- Planillas de calculo.
- Procesadores de palabra.
- Graficos de negocios.
- Bases de datos (d Base II, etc.)
- Contabilidad general, sueldos, y jornales, costos, etc., desarrollados bajo CP/M en Basic, Cobol, Pascal o C.

Con la posibilidad de conexion a linea telefonica permite la transferencia y consulta de datos entre computadoras personales, profesionales o bancos de datos.
La grabacion de archivos es en formato MS-DOS, haciendola compatible con las computadoras profesionales.

## DIDACTICA

Dispone de tres lenguajes para la enseñanza de computacion: LOGO como lenguaje de induccion para los mas chicos. Lenguaje de Programacion en castellano, para todos los que quieran aprender a programar sin conocimientos previos. Y Basic MSX como lenguaje profesional. Mas una amplia variedad de perifericos como el Mouse, Lapiz Optico, Tableta grafica, Track-ball, etc.

## DIVERTIDA

La mas genial para Video-Juegos. Por la amplisima biblioteca de programas –**todos nuevos**– de la norma MSX en el mundo. Y ademas, el Basic MSX permite al usuario generar sus propios juegos con un manejo tan simple, como sólo TALENT MSX puede ofrecer.

## CARACTERISTICAS TECNICAS

- Memoria principal 64 KB ampliable hasta 576 KB.
- Memoria de video: 16 KB RAM.
- ROM incorporada de 32 KB con el MSX-Basic de Microsoft.
- Graficos completos, hasta 32 sprites y 16 colores simultaneos.
- Generador de sonido de 3 voces y 8 octavas.
- Conexion para cualquier grabador.
- Interfaz para salida impresora paralela.
- Conectores para cartuchos y expansiones.
- Fuente para 220 V y modulador PAL-N incorporado.

**DISTRIBUIDORES AUTORIZADOS: CAPITAL FEDERAL:** AMATRIX, Bolívar 173 - ARGECINT, Av. de Mayo 1402 - BAIDAT COMPUTACION, Juramento 2349 - COMPUPRANDO, Av. de Mayo 965 - COMPUSHOP, Córdoba 1464 - COMPUTIQUE, Córdoba 1111, E. P. - COMPUTRONIC, Viamonte 2096 - CP67 CLUB, Florida 683, L. 18 - DALTON COMPUTACION, Cabildo 2283 - ELAB, Cabildo 730 - MICROSTAR, Callao 462 - Q.S.P., Bartolomé Mitre 864 - SERVICIOS EN INFORMATICA, Paraná 164 - DISTRIBUIDORA CONCALES, Tucumán 1458 - MICROMATICA, Av. Pueyrredón 1135 - **ACASSUSO:** MICROSTAR ACASSUSO, Eduardo Costa 892 - **AVELLANEDA:** ARGOS, Av. Mitre 1755 - **BOULOGNE:** COMPUTIQUE CARREFOUR, Bernardo de Irigoyen 2647 - **CASTELAR:** HOT BIT COMPUTACION, Carlos Casares 997 - **LANUS:** COMPUTACION LANUS, Caaguazú 2186 - **LOMAS DE ZAMORA:** ARGESIS COMPUTACION, Av. Meeks 269 - **MARTINEZ:** VIDEO BYTE, Hipólito Yrigoyen 32 - **RAMOS MEJIA:** MANIAC COMPUTACION, Rivadavia 13734 - **SAN ISIDRO:** FERNANDO CORATELLA, Cosme Beccar 249 - **VICENTE LOPEZ:** SERVICIOS EN INFORMATICA, Av. del Libertador 882 - **BAHIA BLANCA:** SERCOM, Donado 327 - SUMASUR, Alsina 236 - **LA PLATA:** CADEMA, Calle 7 Nº 1240 - CERO-UNO INFORMATICA, Calle 48 Nº 529 - **MAR DEL PLATA:** FAST, Catamarca 1755 - **NECOCHEA:** CAFAL, Calle 57 Nº 2920 - SERCOM, Calle 57 Nº 2216 - **TRENQUE LAUQUEN:** COMPUQUEN, Villegas 231 - **CORDOBA:** AUTODATA, Pasaje Santa Catalina 27 - TECSIEM, Santa Rosa 715 - **ROSARIO:** 2001 COMPUTACION, Santa Fe 1468 - MINICOMP, Maipú 862 - SISOR, Urquiza 1062 - **SANTA FE:** ARGECINT, P. San Martín 2433, L. 36 -SISOR, Rivadavia 2553- INFORMATICA, San Gerónimo 2721/25 - **VILLA MARIA:** JUAN CARLOS TRENTO, 9 de Julio 80 - **MENDOZA:** INTERFACE, Sarmiento 98 - BIT & BYTE, 9 de Julio 1030 - **COMODORO RIVADAVIA:** COMPUSER, 25 de Mayo 827 - **GENERAL ROCA:** DISTRIBUIDORA VECCHI, 25 de Mayo 762 - **LA PAMPA:** MARINELLI, Pellegrini 155 - **NEUQUEN:** MEGA, Perito Moreno 383 - EDISA, Roca esq. Fotheringham - **RIO GRANDE:** INFORMATICA M & B, Perito Moreno 290 - **SAN CARLOS DE BARILOCHE:** L. ROBLEDO & ASOCIADOS, Elfein 13, Piso 1º - **TRELEW:** SISTENOVA, Sarmiento 456 - **PARANA:** MARIO GARCIA, Laprida y Santa Fe - **POSADAS:** CENTRO DE COMPUTOS ELDORADO, Colón 2429 - **RESISTENCIA:** FRANCO SANTI, Carlos Pellegrini 761 - **SAN**

# STAFF


LOAD MSX

**Director Periodístico**

Fernando Flores

**Secretario de Redacción**

Ariel Testori

**Redacción**

Fernando Pedró

**Arte y Diagramación**

Fernando Amengual y Tamara Migelson

**Departamento de Avisos**

Oscar Devoto y Nelzo Capello

**Departamento de Publicidad**

Dolores Urien

**Servicios Fotográficos**

Image Bank, Oscar Burriel y Victor Grubicy

**EDITORIAL PROEDI**

**Presidente**

Ernesto del Castillo

**Vicepresidente**

Cristian Pusso

**Director Titular**

Javier Campos Malbrán

**Director Suplente:**

Armengol Torres Sabaté

Load Revista para usuarios de la norma MSX es una publicación mensual editada por Editorial PROEDI S.A., Paraná 720, 5° Piso, (1017) Buenos Aires. Tel.: 46-2886 y 49-7130. Radiollamada: Tel.: 311-0056 y 312-6383, código 5941. Registro Nacional de la Propiedad Intelectual: E. T. M. Registrada. Queda hecho el depósito que indica la Ley 11.723 de la Propiedad Intelectual.



ISSN 0326-8241

Impresión: Calcotam, Fotocromo tapa: Columbia. Fotocomposición: Interamericana Gráfica.

Los ejemplares atrasados se venderán al precio del último número en circulación.



Distribuidor en Capital: Martino, Juan de Garay 358, P. B. Capital.

Distribuidor interior: D G P: Hipólito Yrigoyen 1450, Capital Federal. T.E. 38-9266/9800.

CORREO ARGENTINO CASA CENTRAL

FRANQUEO PAGO CONCESION N° 2538

TARIFA REDUCIDA CONCESION N° 1304


# SUMARIO



# NOTICIAS MSX

## NUEVO SISTEMA OPERATIVO

La empresa LOGICIEL está a punto de lanzar al mercado el SERIUS, un sistema operativo de interfase gráfica.
Este sistema, del tipo del GEOS o el GEM, es totalmente manejado mediante menúes y, según nos informaron, va a estar disponible en dos versiones.
Una para ser usada con joystick y/o teclado, y otra con mouse.
El sistema cuenta con herramientas de trabajo como reloj, calculadora, apuntador, etcétera. Del menú principal se puede ir a la opción ARCHIVO, para abrir, cerrar, renombrar, imprimir o duplicar cualquiera de los archivos.
Además, el menú dispone de otras muchas opciones, por ejemplo, la presentación de los ficheros puede hacerce por nombre, por medida, por fecha o por tipo.
En los próximos números ampliaremos la información sobre este sistema operativo.

## LAPIZ DE PLATA AL DISEÑO

El jurado internacional de la Bienal de diseño de Buenos Aires '87 otorgó el "Lápiz de Plata" al Estudio DIP, de Ricardo Lafon e Isidro Carbonera, por el diseño de la computadora TALENT TPC 310; el modem MN 510 y el teclado numérico TNX 410.
El premio fue asignado a los diseñadores de los equipos de computación Talent, junto con estudios de diseño industrial de nuestro país, de Brasil, de Italia y de Dinamarca.




## ASESORAMIENTO INFORMATICO

Un convenio con el Ministerio de Educación y Cultura de la Provincia de Santa Cruz y otro similar con la Provincia de Santiago del Estero fueron concretados por el CEDI ( Centro para el Desarrollo de la Inteligencia), en conjunto con Talent Computación.
Dichos convenios establecen el compromiso de asesoramiento para incorporar la informática al ámbito educativo y realizar talleres de sensibilización en informática y seminarios de formación de multiplicadores docentes en informática y educación para docentes de todos los niveles y modalidades de ambas provincias.

## TERCER CONCURSO DE PROGRAMAS

El 30 de Noviembre pasado fue la fecha de cierre de la recepción de trabajos para el tercer concurso de programas auspiciado por Telemática S.A.
Recordemos que el primer premio consiste en un periférico a elección entre un monitor, una disquetera y una impresora.
Además el ganador obtendrá una beca para el Departamento de Investigación y Desarrollo de Telemática S.A.
Quien alcance el segundo premio también podrá elegir un periférico entre los antes mencionados.
El jurado ya comenzó a ver los trabajos presentados y para el próximo número tendremos el nombre de los ganadores.

## GANADORES DEL CONCURSO MENSUAL Y TRIMESTRAL

El concurso mensual LOAD MSX llegó a su fin con la entusiasta participación de nuestros lectores.
El ganador de este mes es **Gonzalo Diego Peña**, que vive en Caseros, provincia de Buenos Aires.
Gonzalo nos envió una nota sobre Robots que publicaremos próximamente.
El jurado tuvo que determinar también el ganador del trimestre. En este caso la decisión recayó sobre **Miguel Luis Kopec**, ganador del mes de octubre , cuyo artículo sobre gráficos en la MSX hemos publicado en nuestro último número. También nos envió la nota "Desnudando programas", publicada en otras páginas de esta revista.
Miguel recibirá como premio un software de aplicación, con su respectivo manual, a elección entre el MSX LOGO, el MSX PLAN y el MSX WRITE,que será provisto por Telemática.

en diversas opciones y computadoras.

## CONCRECIONES

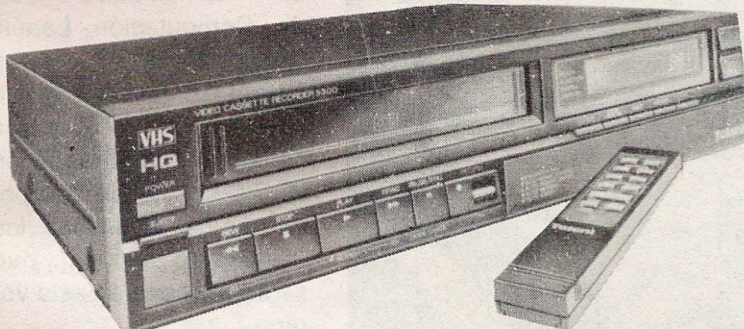


En la Escuela Nacional de Comercio Nº5 "Gral. José de San Martín" se festejó la finalización del Seminario Taller de Formación de Docentes Multiplicadores en Informática y Educación Módulo 1, coordinado por el CEDI (Centro para el Desarrollo de la Inteligencia).

## BANCO DE DATOS

Telemática, la empresa que poduce las computadoras Talent MSX, tiene a su cargo la implementación del banco de datos del Automóvil Club Argentino. Acamática, tal el nombre del servicio, dará información a los socios sobre diversos temas: estados de los caminos, mercado automotor, bursátil y de cambio, hoteles, transportes, etc.

## VIDEO GRABADOR




Un moderno equipo de Video Grabador Reproductor, fue presentado por Talent.
Se trata del Talent 5300, que posee 3 velocidades de grabación, capacidad de 110 canales VHF y UHF, grabación programada de 6 eventos en 14 días, grabación de temporización inmediata (un toque graba 30 minutos hasta 240 minutos), encendido de interferencia, control remoto de 14 posibilidades y búsqueda visual en avance y retroceso.
Este Video Grabador Reproductor se incorpora a la línea de electrónica de Talent, que produce también en el país televisosres, equipos de audio

## EXPO DISNEY

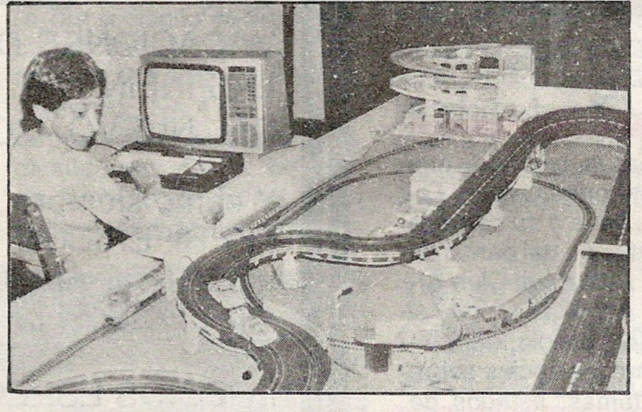
Organizado por **Inforexco** y producido por **Genea**, entre el 4 y 20 de diciembre se lleva a cabo Expo Disney Navidad, Entre sus stands presenta una especial nave espacial. Para que los pequeños usuarios se sientan en una nueva era, **Telemática S.A.** equipó esa nave con muchas computadoras. Sin duda, a través de Mickey, Donald, Blancanieves y ahora la informática, el espíritu de Disney rondará en Autopista Center.

# COMO COMANDAR TRENES Y AUTOS

***Aprenderemos cómo manejar a través de la Talent MSX, trenes de cualquier escala y modelo como también autos eléctricos.***

Debido a que en los últimos tiempos ha sido notable el incremento (en calidad y en cantidad) de la utilización de robots en los sectores más peligrosos y agotadores de la industria, es casi una necesidad empezar a conocer su técnica.

En el Centro de Asistencia al Usuario de Talent MSX de Lanús, cuentan con una hermosa maqueta en donde se demuestra a todo el público, cómo se pueden comandar simultáneamente una carrera de autos, dos trenes eléctricos, un helicóptero y la iluminación de todo un escenario. Todos los elementos utilizados se obtienen fácilmente en jugueterías. El criterio fundamental que promueve este desarrollo es ingresar en forma práctica y sencilla al mundo de la robótica. La computadora controla los coches y demás aparatos de dos formas: enviando mensajes a los motores para ponerlos en marcha y pararlos, y recibiendo mensajes que dicen a la computadora lo que está haciendo el aparato. Los mensajes de entrada y salida pasan por un circuito electrónico llamado interfase. En esta oportunidad veremos cómo enviar mensajes.



Todo esto se puede realizar a través de una interfase que el propio alumno puede desarrollar en clase o en su casa, teniendo la posibilidad de utilizarla en una amplia gama de aplicaciones, incluyendo el más exótico robot, cuya única limitación de funcionamiento es la imaginación.

En este caso en particular se comandarán 2 autos, 2 trenes, 1 helicóptero y, por último, las luces del escenario.

Funcionamiento: para enviar un dato a la salida de la interfase, es necesario generar un programa en BASIC que detallamos a continuación:

```
10 INPUT 'DATO';N
20 FOR I=1 TO 2000
30 OUT 0,N
40 NEXT I
50 GO TO 10
```

en donde el 'dato' son números que responden a la siguiente tabla:

| | ** | 'DATO' |
|---|---|---|
| Auto rojo | D1 | 1 |
| Auto blanco | D2 | 2 |
| Tren 1 | D3 | 4 |
| Tren 2 | D4 | 8 |
| Helicóptero | D5 | 16 |
| Luces | D6 | 32 |
| | | 63 |

Ejemplo: si queremos hacer funcionar el auto rojo pulsamos el número 1.

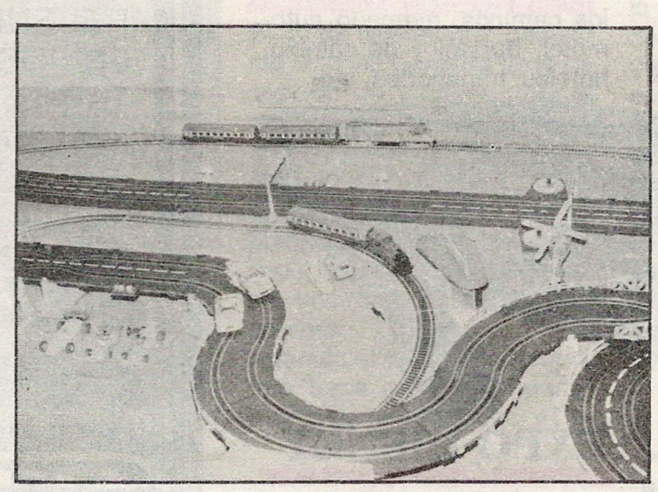
Si queremos encender las luces, pulsamos el número 32. Si queremos que estos hechos se produzcan simultáneamente, pulsamos la suma de ambos, es decir 33 (32+1).

Si pulsamos 63, funcionan todos al mismo tiempo.

## LISTADO DE MATERIALES

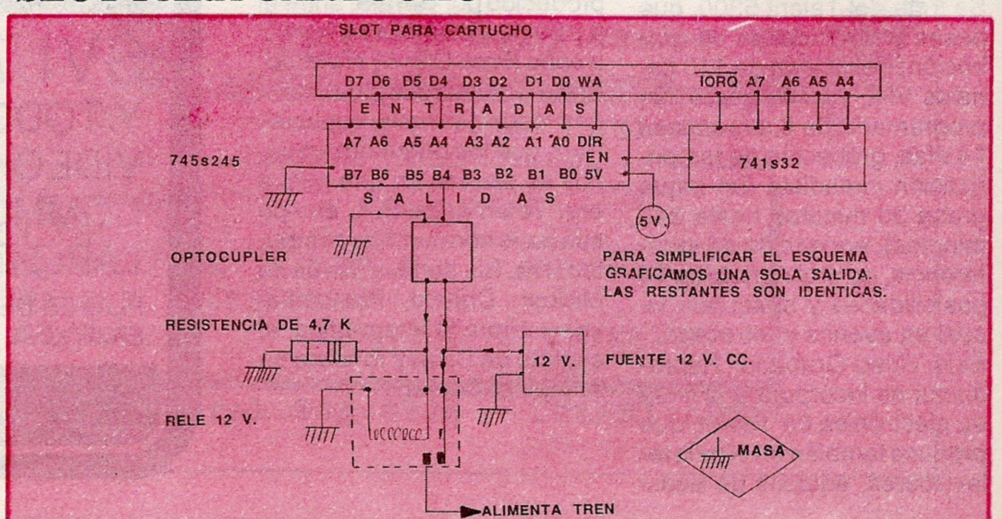
1 Integrado 74LS32
1 Integrado 74LS245
6 Optocupler
6 Relay de automóvil 12 V. CC.
6 Resistencia de 4700 OHM.
1 Fuente de 12 V. CC.
1 Plaqueta de circuito impreso

Hasta aquí tenemos todos los elementos para ponernos a trabajar, es decir, materiales, programa, y el circuito eléctrico. Podemos visitar el local de Computación Lanús, Caaguazú 2186, Lanús Este (TE: 247-0678), en donde a los tradicionales cursos de programación BASIC y Logo, se incorporan en la actualidad los "Cursos de robótica", que serán dictados en este verano.



**= pines del slot de entrada cartucho (ver esquema eléctrico)

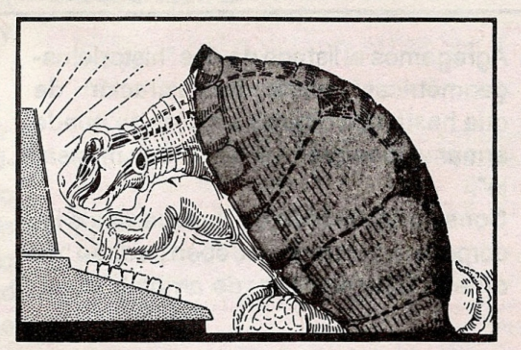
## SLOT PARA CARTUCHO



# ALFABETIZANDO LA TORTUGA

En esta oportunidad, nos ha llegado la colaboración de **Liliana Saidón** y **Liliana Hindi de Cohen**, del **Taller Galileo Galilei**, que es el Centro de Asistencia de Palermo.

Son bien conocidas las amplias posibilidades de graficación en lenguaje Logo a través de la famosa Tortuga Dibujante. En el trabajo en Logo con chicos, sobre



todo con los más grandes, aparece una y otra vez la necesidad de poder agregarles texto a los gráficos de Tortuga. Ya sea para incorporar un "cartel" en un gráfico o para hacer aparecer en pantalla dibujos "tipo historieta", conviene contar con una Tortuga que no sólo dibuje sino que además sepa escribir.

El "procedimiento de apoyo" esct (EScribir con Tortuga) nos permite escribir va-

```
para e :l
hacer "p cursor
fcursor lista redondeo 28 * ( 120 +
coorx ) / 247 redondeo 24 * (70 - coory
 ) / 139
escribir :l
fcursor :p
bon "p
fin

para esct :lv
sp
si vacia? :lv [fy :curs bon "curs parar]
si no nombre? "curs [hacer "curs coory]
e primero :lv
fy coory - 6
esct mp :lv
fin

para f :o :n :l
de :o
repetir :n [ad :l de 360 / :n]
fin

para hist
ot
sp at 30 cp de 90 ad 130 at 230
iz 180
at 53 f 0 5 35
at 50 de 90
repetir 2 [ad 80 de 90 ad 40 de 90]
iz 90
at 50 de 90
repetir 2 [ad 80 de 90 ad 40 de 90]
ad 50 iz 90 sp at 10 de 90
cp f 0 3 26
sp iz 90 ad 50 cp f 90 4 26
sp ad 32 cp
repetir 2 [ad 18 de 90 ad 90 de 90]
sp ad 10 de 90
ad 30 bt
esct [[BA\os]] at 100 iz 90
at 30 esct [[???]]
at 30 iz 90 ad 16 esct [[O]]
at 20 esct [[O]]
ad 8 de 90 at 5 esct [[L]]
at 14
cp repetir 180 [ad 0,1 iz 1]
fin

para hist2
ot
fcolorf 1
sp at 60 cp de 90 ad 130 at 270
iz 180
at 53
f 0 4 25 sp de 45 ad 10 cp fcolorp 7
pintar sp at  10  iz  45  cp fcolorp 15
at 40 f 0 4 25 sp de 45 ad 10 cp fcolorp
 5  pintar  sp at 10 iz 45 cp fcolorp
15 at 160
f 0 4 120
sp de 45 ad 10 cp fcolorp 9 pintar
sp at 10 iz 45 cp fcolorp 15
ad 220
de 90 sp ad 25 de 25 cp ad 12
iz 35 sp ad 20
repetir 3 [bt  esperar  10  esct  [[No
le  hagas]  [caso.Es  un]  [agrandado!!
]]]
```

rios renglones de texto desde donde está "parada" la Tortuga.
Basta "pedirle" a la Tortuga, por ejemplo, esct [[Hola, ¿que tal?] [¿Cómo te va?]] para que aparezcan los dos renglones (Hola, ¿que tal? y ¿Cómo te va?) escritos a partir de donde se encontraba la Tortuga.

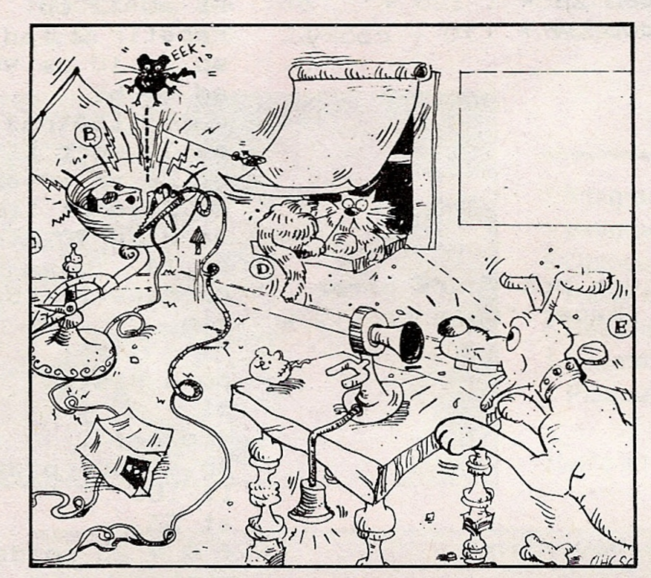
Agregamos el listado de dos "historietas-geométricas" como demostración de que hasta con dibujos simples se puede armar una pequeña historia "con mensaje".
Consejo: los mensajes que queremos incorporar en un gráfico deben pedirse "en orden". Esto significa: de arriba hacia abajo de la pantalla.
Los listados que acompañan esta nota han sido realizados por los alumnos del Taller Galileo Galilei, cuya edad promedio es de 6 años. La forma de utilizar es:
bp hist
bp hist2
fin

# CONTESTANDO LA HOT-LINE

Este mes inauguramos una nueva sección dentro del Rincón que denominaremos "Contestando la Hot-Line". Aquí reflejaremos las inquietudes que surgieron al contestar las llamadas hechas al servicio de consulta telefónica INFOTALENT (38-6601). De esas llamadas, las que nos parezcan más interesantes o aquellas que nos muevan a realizar algún programita o truco serán contestadas en esta columna, para ampliar la breve respuesta telefónica.
Este mes responderemos tres preguntas relacionadas con Talent MSX-BASIC.

**PREGUNTA 1:**
¿Cómo podemos guardar un programa en MSX-BASIC en formato binario, es decir, que se pueda cargar con BLOAD "..",r? Al guardarlo de esta forma podríamos proteger un programa escrito en MSX-BASIC contra listados no permitidos, ya que si se carga el programa , el comando LIST no responde.

**RESPUESTA:**
Para realizar este tipo de protección debemos tener en claro la forma en que el MSX-BASIC almacena un programa en memoria.




Según el MSX Technical Data Book, los programas en MSX BASIC se almacenan a partir de la dirección de memoria &H8000 hasta la posición de memoria en donde se guardan las variables.
Esta posición varía de acuerdo con la longitud del programa BASIC (obviamente). El BASIC guarda dentro de su área de trabajo adonde comienza cada área, y, por lo tanto, el programa de protección almacenará ese valor y cuando se ejecute el programa, la parte escrita en código de máquina se encarga de restaurarlo.
De más está decir que para que la protección sea efectiva tenemos que evitar que el programa pueda ser interrumpido en su ejecución (con las teclas <CTRL> +<STOP >).
Para ello tenemos dos variantes:
1) Usar el comando ON STOP GOSUB ... (ver manual de la consola)
2) Incorporar dentro del programa la siguiente sentencia:
0 POKE &HFBB1,1
Finalmente, la siguiente rutina debe grabarse en código ASCII para poderla cargar -con la instrucción MERGE- junto con el programa. Una vez cargada, se ejecutará con la instrucción GOTO 65000 y aparecerán en la pantalla todos los datos necesarios para poder guardar el programa en el formato BSAVE. Ver listado 2.

## CENTROS DE ASISTENCIA AL USUARIO DE TALENT MSX

**CAPITAL FEDERAL**

Centro Cultural de la Ciudad de Buenos Aires
Taller Logo de computación: Junín 1930
Fundación de Informática y Educación
Centro de Computación Clínica Asistencia al Usuario Discapacitado: Ramsay 2250 - Pabellón F
Tel. 784-2018

Barrio Norte
Uriburu 1063 - Tel. 83-6892/826-6692

Belgrano
Mendoza 2728 - Tel. 781-2271

Centro
Av. Córdoba 654 - Tel. 392-5328/7611/8043/8051/8251

Flores
Gral. Artigas 354 - Tel. 612-3902

Palermo
Guatemala 4733 - Tel. 71-4124

San Telmo
Chile 1345 - Tel. 37-0051 al 54

**GRAN BUENOS AIRES**

Castelar
C. Casares 997 - Tel. 629-2247

Lanús
Caaguazú 2186 - Tel. 247-0678

Morón
Belgrano 160 - Tel. 629-3347

Quilmes
Moreno 609 - Tel. 253-6086 al 89

**PREGUNTA 2:**
¿Qué compatibilidad tiene la Talent MSX2 Turbo en cuanto a software y hardware respecto a la MSX1?
**RESPUESTA:**
La Talent MSX2 Turbo mantiene una total compatibilidad en software y en hardware con la norma MSX versión 1. Esto incluye a todo el software existente (en cualquier lenguaje, incluyendo el código de máquina) y todo el hardware (Unidad de discos, Minilan, Modem, Interfase RS-232, etcétera). La compatibilidad se da en un solo sentido: lo que funciona en MSX 1 funciona en MSX 2 pero el soft que usan las funciones nuevas del MSX 2, no funciona en MSX 1.

**PREGUNTA 3:**
En el juego Chess (Ajedrez), ¿cómo puedo hacer si quiero suspender una partida y volver al menú inicial?
**RESPUESTA:**
Nuestro catálogo de juegos es extenso y contestar por Hot-Line cómo funciona cada juego es un poco problemático. Hemos investigado un poco y llegamos a la siguiente conclusión:
Si pulsamos la tecla <SELECT> cuando nos toca jugar, se pueden ingresar varias opciones, a saber:
<B>: Begin (Inicio).
Reinicia el juego manteniendo el color de cada jugador.
<E>: Empty (Vaciar).
Si contesta <Y> entonces vacía el tablero y aparece el cursor. Si pulsa la barra espaciadora, queda seleccionada la posición y debe ingresar la pieza que desea colocar en notación inglesa: por ejemplo, colocar el rey blanco se ingresa: WK (white king), etcétera.
<L>: Load (Cargar).
Permite cargar desde casete una partida previamente almacenada.
<P>: Put (Poner).
Permite colocar una pieza cualquiera en la posición indicada, siempre que sea válido (por ejemplo, no permite colocar 2 reyes...). La pieza debe nombrarse en notación inglesa.
<S>: Save (Grabar).
Permite grabar en casete una partida en el estado en que se encuentra al momento de ingresar el comando y luego se carga con la opción <L>.
<T>: inTerchange (Intercambiar)
Permite cambiar de lado (el que jugaba con blancas pasa a jugar con negras y viceversa).
<X>: eXecute (ejecutar).
Luego de usar alguna de las opciones anteriores, se le indica a la computadora que juegue su turno.
Por lo tanto, si se quiere cambiar el nivel volviendo al menú inicial, se deben seguir los siguientes pasos:
1) <SELECT><E><Y> (seleccionar vaciado)
2) <SELECT><X> (ejecutar)
Encuentra que perdió alguien y vuelve al menú inicial.

## FIGURA 2

```
65000 ' Esta rutina sirve para grabar el programa en formato
BSAVE
65010 ' Grabarla en formato ASCII para poder "mergearla".
65020 '
65030 DI=INT(PEEK(&HF6C2)+256*PEEK(&HF6C3))
65040 POKE DI+10,&H26:POKE DI+11,PEEK(&H8002):POKE
DI+12,&H2E:POKE DI+13,PEEK(&H8001):POKEDI+14,&H22:POKE
DI+15,1:POKE DI+16,&H80:POKE DI+17,&H21:POKEDI+18,PEEK(&HF6C2)
65050 POKE
DI+19,PEEK(&HF6C3):POKEDI+20,&H22:POKEDI+21,&HC2:POKEDI+22,&HF6:P
OKE DI+23,&HC3:POKEDI+24,&HAC:POKEDI+25,&H73
65060 PRINT"Direcci}n de ejecuci}n:";HEX$(INT(DI+10))
65070 PRINT "BSAVE
(NOMBRE),&H8000,&H";HEX$(INT(DI+30));",&H";HEX$(INT(DI+10))
65080 POKE &H8001,0:POKE &H8002,0:END
```

**CHIMENTOS de juegos:**

Ya que entramos en el tema juegos, vamos a pasar chimentitos jugosos de vez en cuando para que nuestra sección se enriquezca con temas que interesan a muchos usuarios juguetones.
El primer chimento que publicaremos se refiere al juego STARQUAKE. Este es un juego del tipo "aventuras" en donde el personaje central es BLOB, un androide que nos ayuda a cumplir la misión de salvar al planeta Starquake de los designios de los malvados científicos del planeta, que han modificado el reactor nuclear central para que Starquake desaparezca.
El medio más rápido para transportarse dentro del planeta son las cabinas teletransportadoras, pero requieren un código de acceso. Los códigos que hemos podido conocer son los siguientes: SNOOL, UPLAN, ANGOR, VORAX, INDLE, KWAKE (Donde está la central), ANTIO, ERCOT, TARAQ.

# LOS MISTERIOS DEL MSX-DOS.

Por Hugo Daniel Caro

***Quienes deciden tener una disquetera se encuentran con la realidad muy dife - ·ente de la que les ofrece el simple grabador. En esta nota les comentamos vida y obra del sistema operativo de disco. Pondremos luz sobre algunos temas fundamentales.***

Uno de los primeros problemas con que tropieza el usuario de MSX surge cuando decide comprar una unidad de disquetes. Y no es porque dicha unidad presente habitualmente fallas o no funcione bien. Lo que ocurre es que el entorno de trabajo es bastante diferente del que corresponde al uso del grabador con casete, y los términos nuevos que debe manejar el sufrido usuario se incorporan de a poco, a fin de que se pueda utilizar correctamente el nuevo equipamiento.

Si hemos comprado una unidad de discos Talent MSX DPF-550, lo primero que debemos hacer es: "lea cuidadosamente el Manual de Uso".

El mismo ha sido escrito íntegramente en el país (es decir que está en castellano) y hemos tratado de incorporar todos los " chimentos útiles" para el maximo aprovechamiento del equipo.

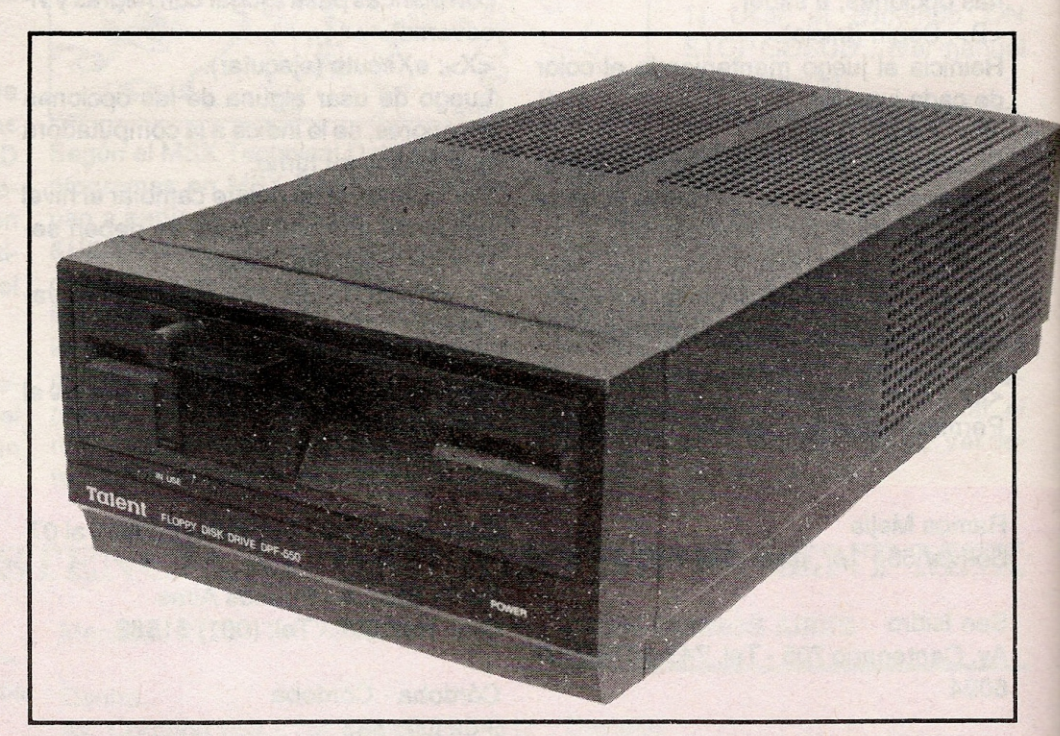
Según la definición que brinda el Manual de Uso, el sistema operativo es un programa que funciona como interfase entre la computadora y el usuario. Su función es facilitar el manejo de las operaciones más comunes que se realizan en una computadora y mantener el control de la misma.

Las tareas que lleva a cabo son:

a) Efectúa las operaciones de entrada/ salida en forma ordenada y consistente, respondiendo a comandos predeterminados. Entre los dispositivos que opera podemos citar: el teclado, unidad de discos flexibles, discos fijos, unidades de cinta magnética, etcétera.

b) Provee también un control de archivos que brinda información sobre el estado de los mismos, longitud, espacio disponible en el dispositivo físico, además de ejecutar los requerimientos de lectura-escritura de un dispositivo determinado.

c) Permite la lectura y ejecución de programas del usuario asignándoles la dirección de comienzo y ejecución, el área de variables propias, las llamadas a rutinas del sistema, etcétera. El sistema operativo, en definitiva, facilita todas las tareas tediosas del sistema, pues si el usuario debiera realizarlas por sí solo, serían extremadamente complicadas.




Por ejemplo, si tuviéramos que mostrar por pantalla el contenido del archivo, deberíamos hacer lo siguiente: buscar el nombre del mismo en el directorio y, una vez ubicado, tomar la posición dentro del disco donde comienza.

Una vez localizada esta posición, se deben transferir byte a byte los datos almacenados en ese cluster, controlar si no finalizó el archivo y rescatar la posición donde se encuentra el siguiente cluster. Etcétera, etcétera, etcétera...

Es mucho más sencillo encomendarle al DOS que nos traiga los datos relacionados a un archivo ingresando solamente: A>TYPE PEPE En esta nota nos vamos a referir a la vida y obra del sistema operativo de disco que trabaja con las MSX: el MSX-DOS. Trataremos de poner luz sobre algunos temas clásicos del sistema: compatibilidades, capacidades, áreas de trabajo, etcétera.

Se requieren dispositivos de gran capacidad de almacenamiento con tiempo de acceso de alta velocidad para aplicaciones comerciales y profesionales. Debido a esto se le incorporó a la norma MSX el sistema operativo de disco. El DOS (disk operating system) sirve además para poder manejar una gran cantidad de información sobre disco en forma efectiva. MSX-DOS es un derivado del MS-DOS que, como todos sabemos, es "el" sistema operativo empleado por las computadoras de 16 bits (IBM PC, etcétera) . Por lo tanto, el MSX-DOS es el sistema operativo más poderoso que se ha desarrollado para computadoras con Z-80.

¿Qué tipo de software es MSX-DOS? ¿Qué ofrece a los usuarios? Vamos a describir y presentar las capacidades, funciones y configuración de software del MSX-DOS.

## PRESENTACION EN SOCIEDAD

**1. Características del MSX-DOS:**

* Consolidación del entorno de sistema operativo de disco. Es el sistema operativo para las computadoras MSX. Trabaja con cualquier versión de las norma y puede funcionar sin problemas ya sea en MSX1 y MSX2. Las operaciones de disco en MSX se realizan siempre vía MSX-DOS. Incluso también el MSX Disk-BASIC utiliza este medio, ya que usa el BDOS para las tareas de entrada/salida. MSX-DOS y Disk BASIC requieren el mismo formato de disco, lo que hace innecesario la conversión de archivos entre BASIC y el DOS. Esto incrementa sobremanera la eficiencia de operación y permite una mejor utilización del recurso de archivo cuando se utiliza el MSX-DOS como un entorno de desarrollo de software.

* Compatibilidad con MS-DOS: El MSX-DOS, -basado en el MS-DOS (versión 1.25) que es un sistema operativo para las computadoras personales de 16 bits-, utiliza el mismo formato de grabación y lectura de archivos que éste. Por lo tanto, ambos son compatibles a nivel de archivos. Esto significa que el MSX-DOS puede leer y escribir archivos almacenados en discos del MS-DOS y viceversa. Ambos sistemas operativos usan comandos similares, lo que hace que los usuarios que están familiarizados con el MSX-DOS puedan utilizar rápidamente el MS-DOS cuando pasan a una máquina más grande, del tipo PC compatible.

* Utilizando aplicaciones en CP/M: El MSX-DOS posee compatibilidad de "llamadas al sistema" con CP/M y puede ejecutar casi todos los programas creados sobre CP/ M sin necesidad de modificarlos. La mayoría de las aplicaciones de CP/M pueden utilizarse con MSX-DOS. Esto abre la posibilidad de utilizar una gran librería de programas ya existentes que se pueden ejecutar en las máquinas MSX.

**2. Entorno MSX-DOS.**

* Requerimientos mínimos: Para usar MSX-DOS se requiere una configuración mínima de 64 kbytes de RAM, un televisor o monitor y una unidad de discos que incorpore el Disk BASIC, entre otras cosas. Las computadoras MSX pueden utilizar el MSX- DOS solo si poseen 64 kbytes de RAM (como la Talent DPC-200). En cambio las computadoras MSX2 siempre tienen 64 kbytes de RAM, ya que la norma así lo requiere. Por lo tanto estas máquinas siempre podrán correr el MSX-DOS.

El software que es necesario utilizar con la unidad de discos siempre viene incorporado a la misma en forma de ROM (como en el caso de la unidad DPF-555).

* Capacidad del sistema: Soporta hasta 8 unidades de disco. Sin embargo, por razones de manejo del control, se pueden conectar 2 unidades por control, lo que hace que en una Talent MSX DPC-200 se puedan utilizar hasta cuatro unidades de disco.

Cuando se usa solamente una unidad de discos, el MSX-DOS tiene la característica de simular 2 unidades (es decir que utiliza un drive como si fueran dos, reemplazando los disquetes temporariamente). Soporta ingreso desde teclado, salida por pantalla y por impresora.

* Soporte de medios de almacenamiento: Posee un administrador de archivos muy flexible que le permite independizarse de la estructura física del disco, y manejar distintos tipos de formato sobre disquetes de 5 1/4 pulgadas o 3 1/2 pulgadas. Pueden utilizarse discos de simple lado, llamados 1DD, o de doble lado, llamados 2DD. Cada uno de ellos puede almacenar formatos de track de 8 o 9 sectores, lo que en definitiva genera 8 tipos diferentes de formatos. Estos son los mismos que utiliza por Microsoft, y en el caso de los discos 5 1/4 doble lado doble densidad, las características son las siguientes:

| | |
|---|---|
| ID medio: | 0EBh |
| Número de lados: | 2 |
| Tracks por lado: | 40 |
| Sectores por track: | 9 |
| Bytes por sector: | 512 |
| Tamaño del cluster (en sectores): | 2 |
| Tamaño del FAT (en sectores): | 2 |
| Número de FATs: | 2 |
| Máxima cantidad de archivos: | 112 |

En una próxima nota pondremos luz sobre los términos FAT, track, sectores, etcétera.

**3. RECURSOS DEL SISTEMA MSX-DOS:**

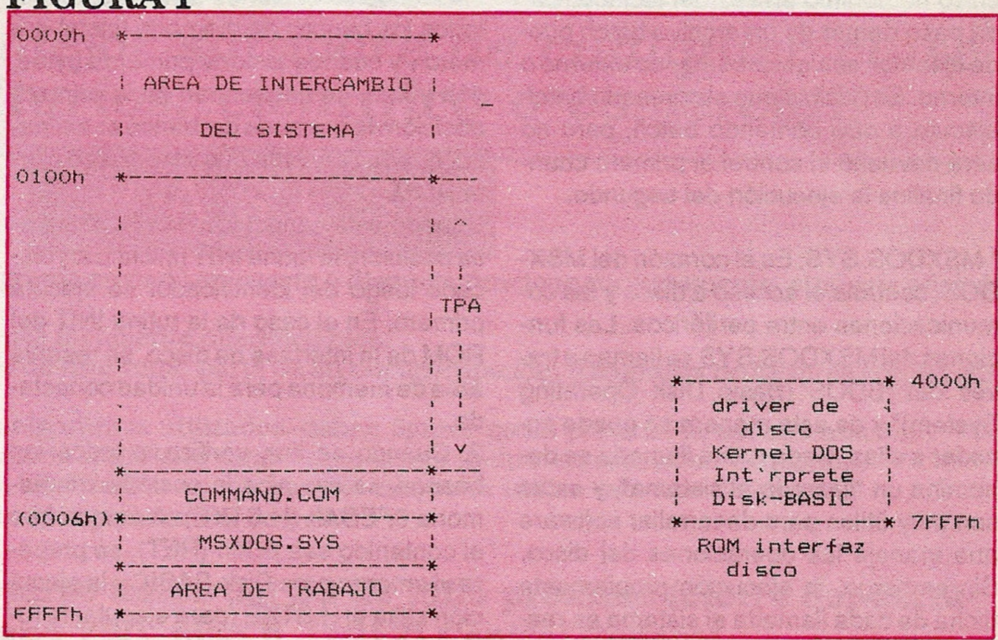
* Mapa de Memoria: El sistema está compuesto por los siguientes módulos:
COMMAND.COM
MSXDOS.SYS
La ROM de la interfase de la unidad de disco.

La configuración de memoria cuando el MSX-DOS está activo es la descripta en la figura 1. COMMAND.COM y MSXDOS.SYS son archivos de discos que se cargan en la memoria en el momento en que se inicializa (en la jerga computacional "bootea") el MSX-DOS. La ROM de la interfaz de disco incluye el driver de disco, el Kernal (núcleo) del DOS y el Intérprete MSX Disk-BASIC.

El área de memoria que abarca desde 00h a FFh de RAM se denomina área de intercambio del sistema, y es utilizada por el MSX-DOS para intercambiar información con otros programas. Es importante para la utilización de las denominadas "llamadas al sistema" que describiremos en detalle en la siguiente nota. El área que comienza en 0100h y termina donde indica el contenido de la posición de memoria 0006h de RAM se denomina TPA (Transient Program Area - Area de Programa Transitoria). Esta es el área adonde accede el usuario. MSXDOS.SYS siempre reside en direcciones superiores al TPA (y cuando se modifica el área donde reside los resultados pueden ser catastróficos). El COMMAND.COM se ubica en el TPA.

* COMMAND.COM.: La operación principal ejecutada por el MSX-DOS es aceptar el ingreso de comandos desde el teclado y ejecutarlos. En este caso, el pro-

**FIGURA 1**

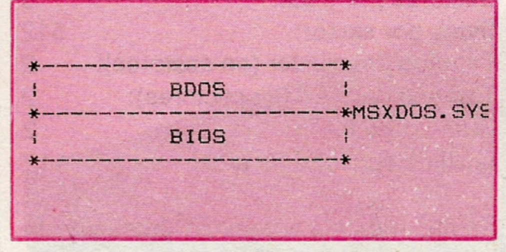


grama COMMAND.COM es el responsable del proceso de obtener una secuencia de caracteres para interpretarla y ejecutarla, o aceptar comandos desde la interfaz del usuario.
Los programas que puede ejecutar el COMMAND.COM son:
- Comandos internos
- Comandos en lotes (batch)
- Comandos externos
Los comandos internos son parte integrante del COMMAND.COM y residen en RAM. Ingresar un comando interno hace que el COMMAND.COM lo llame y ejecute de inmediato .
Para los comandos externos, COMMAND.COM carga la rutina desde el disco al TPA y la ejecuta. Para que así ocurra, el archivo que almacena a dicha rutina debe finalizar con la extensión ".COM" y estar almacenado en código de máquina.
La ejecución de los comandos externos siempre comienza en 100h. En este caso, el COMMAND.COM libera su propia área para el comando externo. Es decir, el COMMAND .COM puede llegar a borrarse a sí mismo y luego escribir el comando externo dentro de sí; cuando un comando externo es lo suficientemente pequeño y no utiliza la parte superior de la TPA, el COMMAND.COM no es destruido.
Cuando el comando externo finaliza con la instrucción en código de máquina 0C9h (RET), el MSXDOS.SYS examina si el COMMAND.COM fue destruido (utilizando una suma de control o checksum) y, si así ocurrió, lo vuelve a cargar en la RAM y le devuelve el control.
Los comandos batch se ejecutan obteniendo una línea de comando desde el archivo homónimo en vez del teclado. Cada paso dentro de un archivo batch puede ejecutar cualquier comando externo o interno. Es posible que el comando batch ejecute a otro comando batch, pero no será devuelto el control al primero cuando finalice la ejecución del segundo.

* MSXDOS.SYS: Es el corazón del MSX-DOS, controla el acceso a disco y las comunicaciones entre periféricos. Las funciones del MSXDOS.SYS se llaman a través del "BDOS (Basic Disk Operating System)" y de esta manera se puede acceder a ellas. Cada rutina llamada se denomina un "llamado al sistema", y estas son muy útiles para desarrollar software que maneje las operaciones del disco. Sin embargo, la ejecución propiamente dicha de cada llamada al sistema es realizada por el Kernal del DOS.
El MSXDOS.SYS es un intermediario que arregla los requerimientos de entrada/salida desde el COMMAND .COM o comandos externos y los pasa al Kernal del DOS. MSXDOS.SYS incluye una porción llamada BIOS diferente del BDOS, como se muestra en la figura 2. La BIOS, que ha sido preparada para ser compatible con CP/M, normalmente no se utiliza.

* Kernal del DOS: Son las rutinas fundamentales de entrada/salida y residen en la ROM de la interfase de disco. Ejecutan las funciones del BDOS del MSXDOS.SYS. En realidad, todas las funciones de "llamada al sistema" pueden ejecutarse utilizando solo el Kernal del DOS. Disk-BASIC ejecuta "llamadas al sistema" invocando en forma directa al Kernal.
* Procedimiento para invocar al MSX-DOS: Vamos a describir técnicamente cuáles son los procedimientos que sigue el sistema MSX para reconocer al MSX-DOS.
Suponemos que el que lee los siguientes párrafos conoce el entorno MSX y su utilización a nivel código de máquina.
Cuando se enciende la máquina con la unidad de discos conectada, conteniendo

## FIGURA 2: MSXDOS.SYS




un disquete, los pasos seguidos por la computadora son los siguientes:
1. Cuando se enciende la MSX examina todos los bancos de memoria (slots) primero , y cuando los dos primeros bytes, 41h y 42h, se encuentran en la primera posición de memoria del banco examinado, queda señalado que este posee cierta ROM.
Cuando está conectada una ROM en esa posición, la rutina INIT (inicializar) ubicada luego del identificador se ejecuta primero. En el caso de la rutina INIT del ROM de la interfase de disco, se reserva área de memoria para la unidad conectada.
2. Cuando se han verificado todos los bancos, se examina la posición de memoria 0FEDAh (H.STKE). A menos que el contenido sea 0C9h (RET), se prepara el entorno para Disk-BASIC y la ejecución salta al H.STKE (para ejecutar la rutina de inicialización que fue incorporada en H.STKE).
3. Cuando el contenido de H.STKE ha sido C9h en el examen anterior, se busca si existe algún cartucho con la entrada TEXT (es decir, que tenga un programa en MSX-BASIC escrito en ROM) y, si se encuentra, se prepara el entorno del Disk-BASIC , para luego ejecutarse el programa BASIC en ROM.
4. Después, si lo anterior no se verificó, el contenido del sector lógico 0 (normalmente llamado "Boot sector") se transfiere de 0C000h a 0C0FFh. A esta altura, si ocurre un error "Drive not ready" (unidad sin disquete) o "Read error" (error en lectura) o el primer byte del sector no es 0EBh o 0E9h, se invoca el Disk-BASIC.
5. Se llama a la rutina 0C01Eh con el "Carry" apagado.
Normalmente, dado que el código "RET NC" se encuentra almacenado en esta posición, no se ejecuta nada y el control vuelve al sistema. Todo programa escrito en código de máquina se invoca de esta manera automáticamente.
6. Se examina la capacidad de RAM (sin destruir su contenido). Si la computadora tiene menos de 64 kbytes se invoca al Disk-BASIC.
7. Se prepara el entorno MSX-DOS y se llama a 0C01Eh con "Carry" encendido. Luego, el MSXDOS.SYS se carga desde 100h, y la ejecución salta a 100h. Después de esto, el MSXDOS.SYS se autotransfiere a las posiciones más altas de memoria. Si no existe el MSXDOS.SYS, se invoca al Disk-BASIC.
8. MSXDOS.SYS carga al COMMAND.COM en 100h y salta a esta dirección. El COMMAND. COM también se autotransfiere a una posición de memoria más alta y luego comienza su ejecución. Si el COMMAND.COM no existe en el disco, se muestra el mensaje " Insert a Diskette" (inserte un disco...) y la ejecución espera que se inserte el disco con el COMMAND.COM.
9. Cuando se carga por primera vez el MSX-DOS, si existe un archivo llamado "AUTOEXEC.BAT" se ejecuta como un archivo batch. Si no se invoca el MSXDOS.SYS y se llama al Disk-BASIC, si existe un programa BASIC almacenado como AUTOEXEC.BAS, el mismo se carga y se ejecuta. En una próxima nota describiremos en detalle el formato de almacenamiento de discos (¡que también les servirá a los usuarios del MS-DOS!) y las famosas "llamadas al sistema" que tanto simplifican la creación y ejecución de rutinas que manejan el almacenamiento de datos en disco.

# 4to CONCURSO

# DE PROGRAMAS

**auspiciado por TELEMATICA S.A. que proveerá los siguientes Premios:**

## PRIMER PREMIO

**UN PERIFERICO**

(a elección entre un monitor, y una disquetera y una impresora).

## SEGUNDO PREMIO

**UN PERIFERICO**

(a elección entre un monitor, una disquetera y una impresora).

## ESPECIAL

Entre los programas recibidos, algunos de ellos podrán ser editados por SYSTEMAC S.A., reconociéndose los derechos de autor.

***Se premiará el mejor software de cualquier clase (juegos, utilitarios, científico o comercial).***

## BASES

No sólo será indispensable que el programa enviado en casete ó disquete funcione correctamente, sino que además debe cumplir con ciertas reglas:

.Programación estructurada en bloques fácilmente diferenciables.

.Fácil seguimiento del mismo y detalle de éste como parte de su documentación. (Diagrama de bloques con los números de línea que los identifiquen).

.Aclaración y clara explicación de los algoritmos utilizados, deben figurar como parte de la documentación.

.Las variables y/o direcciones de memoria utilizados también se deben incluir en esta documentación.

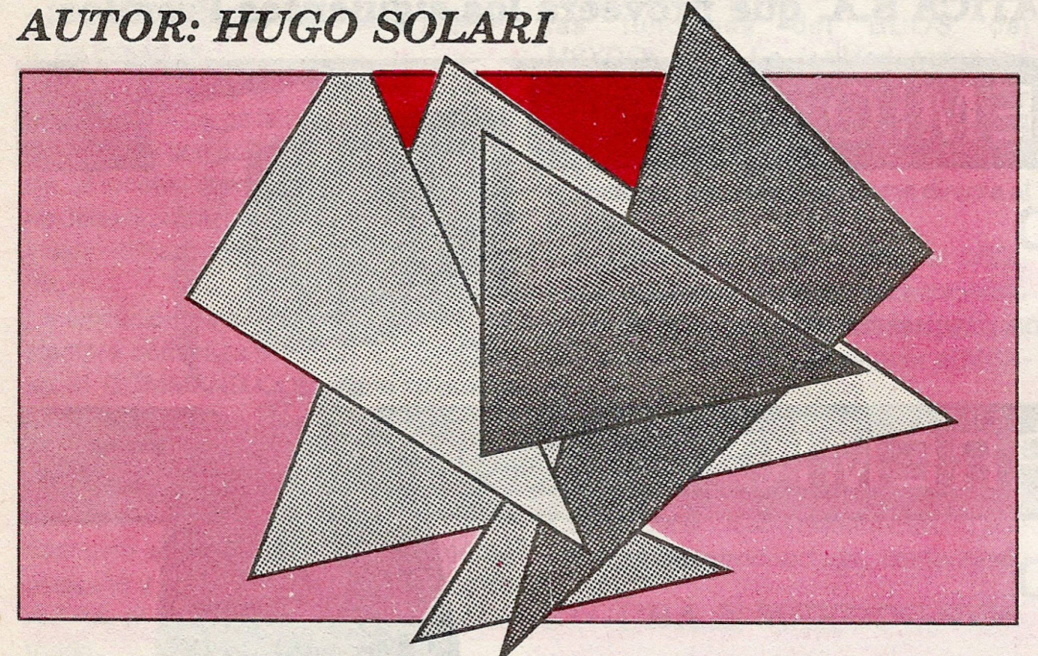
.Listado de nemónicos assembler y la localización en memoria si es que se utiliza este tipo de lenguaje.

.Calidad y originalidad de gráficos, sonidos y pantallas de menú.

**Los trabajos deberán enviarse antes del 30 de julio próximo (cierre del certámen) a: Paraná 720, piso 5, (1017) Capital Federal.**

# RESOLUCION DE TRIANGULOS OBLICUANGULOS

## CLASE: UTILITARIO/EDUCATIVO
## AUTOR: HUGO SOLARI



El objetivo de este programa es encarar la resolución de triángulos (planos) oblicuángulos para la aplicación por parte del docente o estudiante de trigonometría, como también en la labor profesional del ingeniero, topógrafo o agrimensor.

Se pueden presentar cinco casos diferentes según los datos disponibles y las incógnitas por resolver. Los mismos los podemos ver sintetizados en el cuadro número 1.

Por una razón de practicidad se ha utilizado una nomenclatura en el programa y otra, la que se aplica habitualmente, en el menú gráfico del cuadro número 2.

Cuando por alguna razón ingresamos datos incorrectos, que no cumplen con los requisitos exigidos, la computadora nos lo hará saber con un mensaje de tres líneas . La primera es común a todos los casos, al igual que la tercera, que está recuadrada. La segúnda nos da la característica del problema que surgió.

## REQUISITOS DE DATOS HA INGRESAR

a) Lados: deben ser valores positivos menores a 10.000.000 y pueden contar hasta con dos decimales. En caso de que se ingresen los tres lados como datos, se puede producir un mensaje de error si aquellos no cumplen una relación que permita formar un triángulo.

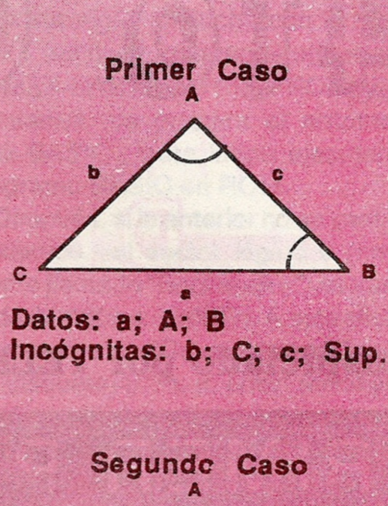
b) Angulos: los valores de los ángulos deben ser menores a 180G, y deben ser introducidos en el sistema sexagesimal. En los casos en que se ingresen como datos dos ángulos, también se debe cumplir la condición de que la suma de ambos sea menor a 180G.

## COMO FUNCIONA EL PROGRAMA

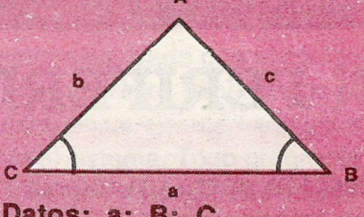
El programa comienza construyendo en alta resolución el menú gráfico que contiene las cinco opciones de cálculo, indicadas con su número de caso dentro del triángulo correspondiente.

Los elementos conocidos, los datos, en color blanco, y las incógnitas en color negro. La superficie es incógnita en todos los casos. La selección de los mencionados colores responde a dos razones: una es la posibilidad de obtener un gráfico claro aún utilizando un monitor monocromático o TV blanco y negro, y la se-

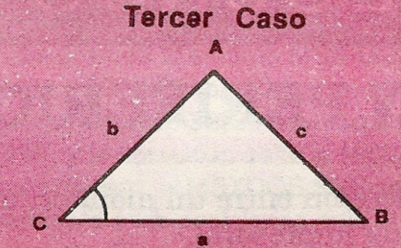
Cuadro 1



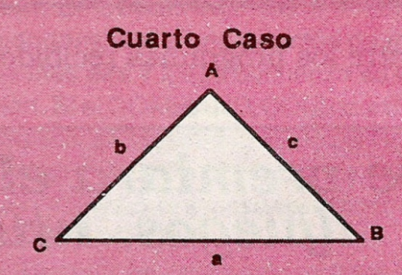

Datos: a; A; B
Incógnitas: b; C; c; Sup.



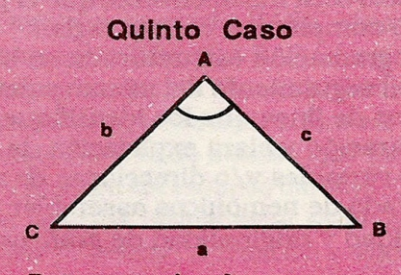

Datos: a; B; C
Incógnitas: A; b; c; Sup.




Datos: a; b; C
Incógnitas: A; B; c; Sup.

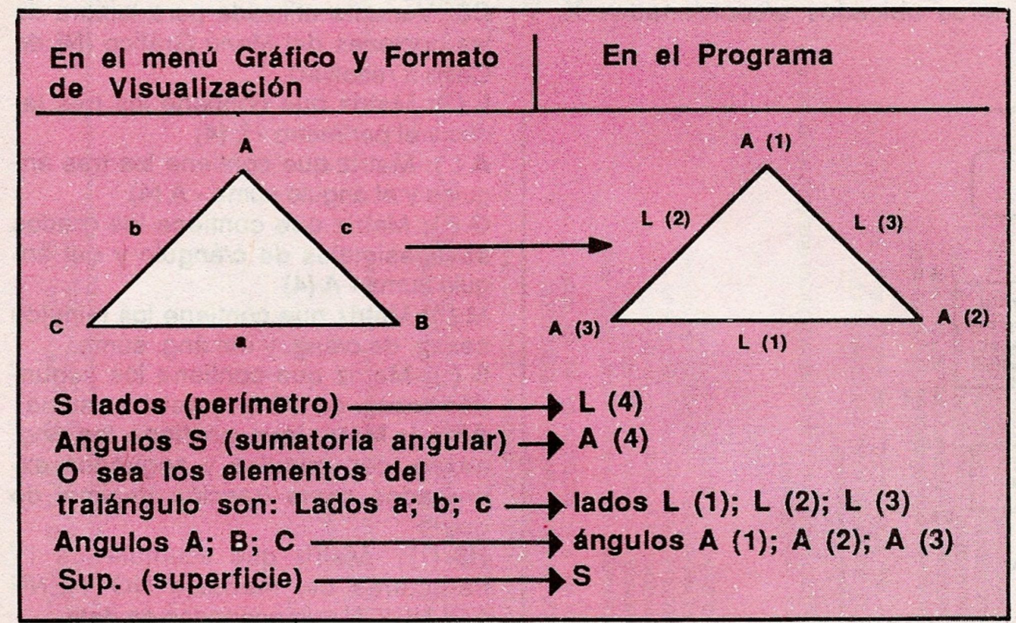



Datos: a; b; C
Incógnitas: A; B; C; Sup.




Datos: a; b; A
Incógnitas: B; C; c; Sup.

Cuadro 2




gunda es de orden didáctico, ya que el color blanco nos da la idea de lo conocido y el negro nos sumerge en las profundidades de lo desconocido.

Una vez ingresada la opción de cálculo aparece una nueva pantalla que nos indica el número de casos a resolver y, en forma cóncisa, los datos y las incógnitas del problema, para inmediatamente pasar a pedirnos la introducción de los elementos conocidos. Tipeado el último dato y asimilado por la computadora, el proceso de cálculo se ejecuta rápidamente y aparece la respuesta. Por ejemplo ver figura 1.

Finalmente nos interroga sobre la posibilidad de realizar un nuevo cálculo, y queda a la espera de la introducción de la respuesta.

## CONSIDERACIONES FINALES

Por razones de redondeo se puede generar un error mínimo en la suma de los ángulos. Esto lo indica la computadora como lo vemos en el siguiente ejemplo de la figura 2.

En el quinto caso se pueden presentar tres variantes con relación al resultado. Ellas son:

1. Que el problema tenga una única solución, situación que se nos indica.
2. Que tenga dos soluciones. La segunda solución se obtiene tipeando un "2", y también se nos indica de tal posibilidad.
3. Que no tenga solución y se genere el mensaje sonoro-visual "NO TIENE SOLUCION ". Al cabo de unos instantes nos propone ingresar nuevos datos, y continúa en la forma ya indicada.

## ESTRUCTURA DEL PROGRAMA

**80:** Dimensiona variables
**90-110:** Define y denomina funciones del usuario. Limpia la pantalla. Asigna valores a las constantes PI y LS (límite superior) y PY
**120-130:** Prepara la pantalla
**140-390:** Menú gráfico
**400-420:** Espera opción. Visualiza "cursor flecha" bicolor
**440:** Acepta la opción elegida y ramifica según caso a resolver
**460-600:** Resuelve el Primer Caso
**620-770:** Resuelve el Segundo Caso
**790-970:** Resuelve el Tercer Caso
**990-1160:** Resuelve el Cuarto Caso
**1180-1580:** Resuelve el Quinto Caso
**1600-1800:** Rutina de Visualización
**1810:** Limpia la pantalla. Finaliza
**1830-1890:** Subrutina de conversión angular (radianes a sexagesimales)
**1910-1960:** Subrutina para visualización de un mensaje sonoro-visual
**1970-2010:** Subrutina para ingreso de datos por teclado
**2020-2040:** Subrutina que contiene algoritmos comunes a varios casos de resolución
**2050:** Subrutina de reatardo, donde RE= cadencia
**2060-2100:** Subrutina que permite pulsando la barra espaciadora reingresar datos. Recuadra el mensaje
**2110-2120:** Subrutina para generar el "cursor flecha" en doble formato
**2130:** Subrutina separador gráfico
**2140-2150:** Subrutina para recuadrar el formato de visualización y mensaje "Presione la barra espaciadora para reingresar datos"
**2160-2200:** Subrutina con mensajes característicos
**2220-2250:** Datas

## VARIABLES UTILIZADAS

**V, W ( ):** Matriz que acumula las coordenadas de 21 puntos del Menú Gráfico en color blanco, almacenadas en data.
**A$ ( ):** Matriz que contiene caracteres según coordenadas V, W, almacenadas en data
**X, Y ( ):** Matriz que acumula las coordenadas de 15 puntos del Menú Gráfico en color negro, almacenadas en data

Figura 1

```
CUARTO CASO
DATOS: a; b; c ~ INCOG.: A; B; C; Sup.

» Lado a = ? 3
» Lado b = ? 4
» Lado c = ? 5

Lado a=     3.00   Ang. A= 36° 52' 12"
Lado b=     4.00   Ang. B= 53° 7' 48"
Lado c=     5.00   Ang. C= 90° 0' 0"

Lados=      12.00   Ang.= 180° 0' 0"
SUPERFICIE=        6.00

¿Resolverá otro triángulo? ➔ S/N
```

Figura 2

```
CUARTO CASO
DATOS: a; b; c ~ INCOG.: A; B; C; Sup.

» Lado a = ? 4.5
» Lado b = ? 6
» Lado c = ? 7

Lado a=       4.50   Ang. A= 39° 34' 16"
Lado b=       6.00   Ang. B= 58° 8' 40"
Lado c=       7.00   Ang. C= 82° 17' 3"

Lados=        17.50   Ang.= 180° 0' 0"
Error de redondeo=  -1"
S (A)=16.4  S (B)=40.4  S (C)=3.2
SUPERFICIE=        13.38

¿Resolverá otro triángulo? ➔ S/N
```

## DIAGRAMAS DE FLUJO

Comienzo

80
Dimensionamiento def. y den. funciones Asigna val. const. Prepara pantalla
130

140
Menú gráfico
390

400
C$ opción
1 2 3 4 5
440

460 1º caso 600
620 2º caso 770
790 3º caso 970
990 4º caso 1160
1180 5º caso 1580

1600
Comienza Rutina de visualización
Resultado
1730

1740
VAL (C$)=s and C < L (1) and L (1) < L (2) and A (2) < PI/2
No
SI
goto 1520

1750
¿Resolverá otro triángulo? [s/n]
1800
No
SI
Run
Termina rutina visual

1810
Fin

**B$ ( ):** Matriz que contiene 15 caracteres ubicados s/coordenadas X, Y almacenadas en data

**C$:** Variante utilizada para lectura de las opciones del Menú Gráfico (Nº de Caso a resolver)

**L ( ):** Matriz que contiene los tres lados y el perímetro - L (4)

**A ( ):** Matriz que contiene los tres ángulos y el ángulo suma - A (4)

**G ( ):** Matriz que contiene los grados sexagesimales de c/ángulo y del ángulo suma - A (4)

**M ( ):** Matriz que contiene los minutos sexag. de c/áng. y del áng. suma

**S ( ):** Matriz que contiene los segundos sexag. de c/áng. y del áng. suma

**AS ( ):** Matriz que contiene los áng. sexag. con minutos y segundos expresados como fracción decimal de grado

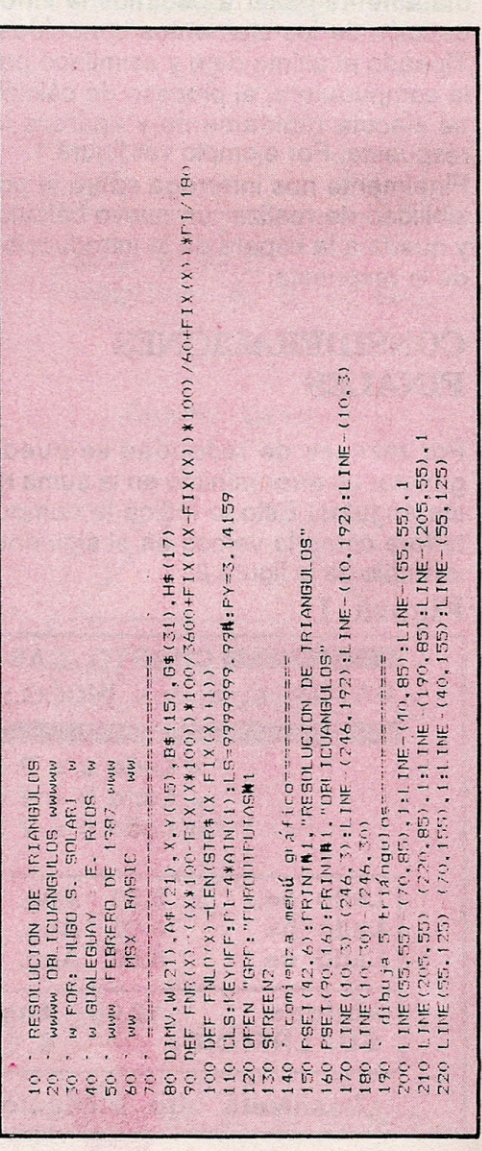
**H$ ( ):** Matriz que acumula los 17 caracteres del mensaje NO TIENE SOLUCION, almacenados en data

**D$:** Variable que toma la decisión de recalcular (Quinto Caso)

**E$ ( ) y F$ ( ):** Matrices empleadas en la rutina de visualización que acumula cadena de caracteres alfanuméricos, almacenados en data

**I$:** Variable utilizada para la leetura de las respuestas S/N o s/n

**J$:** Variable que controla si la barra espaciadora es pulsada



```
10 ' RESOLUCION DE TRIANGULOS
20 ' wwww OBLICUANGULOS wwwww
30 ' w FOR: HUGO S. SOLARI  w
40 ' w GUALEGUAY   E. RIOS  w
50 ' www  FEBRERO DE 1987 www
60 ' ww    MSX  BASIC     ww
70 ' ================================
80 DIMV,W(21),A$(21),X,Y(15),B$(15),G$(31),H$(17)
90 DEF FNR(X)=((X*100-FIX(X*100))*100/3600+FIX((X-FIX(X))*100)/60+FIX(X))*PI/180
100 DEF FNLO(X)=LEN(STR$(X-FIX(X)+1))
110 CLS:KEYOFF:PI=4*ATN(1):LS=9999999.99#:PY=3.14159
120 OPEN "GRP:"FOROUTPUTAS#1
130 SCREEN2
140 ' comienza menú gráfico===========
150 PSET(42,6):PRINT#1,"RESOLUCION DE TRIANGULOS"
160 PSET(90,16):PRINT#1,"OBLICUANGULOS"
170 LINE(10,3)-(246,3):LINE-(246,192):LINE-(10,192):LINE-(10,3)
180 LINE(10,30)-(246,30)
190 ' dibuja 5 triángulos=============
200 LINE(55,55)-(70,85),1:LINE-(40,85):LINE-(55.55),1
210 LINE(205,55)-(220,85),1:LINE-(190,85):LINE-(205,55),1
220 LINE(55,125)-(70,155),1:LINE-(40,155):LINE-(55,125)
```

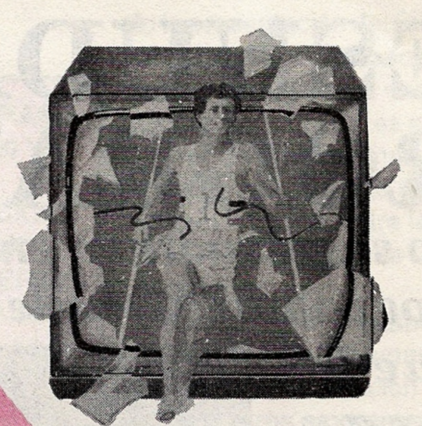
```
230 LINE(130,125)-(145,155):LINE-(115,155):LINE-(130,125)
240 LINE(205,125)-(220,155):LINE-(190,155):LINE-(205,125),1
250 'enumera los 5 casos y designa todos los elementos=====================
260 RESTORE2220
270 FORA=1TO21
280 READV,W:READA$
290 PRESET(V,W):PRINT#1,A$
300 NEXTA
310 COLOR 1
320 RESTORE2230
330 FORA=1TO15
340 READX,Y:READB$
350 PRESET(X,Y):PRINT#1,B$
360 NEXTA
370 PRESET(96,65):PRINT#1,"INCOGNITAS"
380 PSET(95,95):PRINT#1,"Superficie"
390 COLOR 15:PRESET(50,173):PRINT#1,"Ingrese N° de caso a                resolver
    [1/5]" :BEEP:BEEP:BEEF
400 C$=INKEY$
410 COLOR15:PRESET(140,183):PRINT#1,"=▶":RE=200:GOSUB2050
420 IFC$=""ORC$<"1"ORC$>"5"THENCOLOR1:PRESET(140,183):PRINT#1,"=▶":RE=200:GOSUB2
050:GOTO 400
430 ' ================================
440 COLOR15:SCREEN0:WIDTH39:ONVAL(C$)GOTO460,620,790,990,1180
450 ' primer caso====================
460 CLS:PRINT"CCCCCCCCCCC PRIMER  CASO CCCCCCCCCCC"
470 PRINT:PRINT"DATOS: a; A; B # INCOG.: b; c; C; SUP."
480 GOSUB2130
490 GOSUB1970:IFL(1)<=0THENGOSUB1910:GOSUB2160:GOTO 460
500 GOSUB1990:IFLO>7THENGOSUB1910:GOSUB2190:GOTO 460
510 IFA(1)<=0ORA(1)>=PI THENGOSUB1910:GOSUB2170:GOTO460
520 GOSUB2000:IFLO>7THENGOSUB1910:GOSUB2190:GOTO 460
530 IFA(2)<=0ORA(2)>=PITHENGOSUB1910:GOSUB2170:GOTO 460
540 IFA(1)+A(2)>PYTHENGOSUB1910:GOSUB2180:GOTO 460
550 A(3)=PI-A(1)-A(2)
560 GOSUB2020
570 GOSUB2030
580 S=.5*L(1)*L(3)*SIN(A(2)):GOSUB1830
590 IFL(1)>LSORL(2)>LSORL(3)>LSORL(4)>LSTHENGOSUB1910:GOSUB2200:GOTO 460
600 GOTO1600
610 ' segundo caso====================
620 CLS:PRINT"CCCCCCCCCCC SEGUNDO CASO CCCCCCCCCCC"
630 PRINT:PRINT"DATOS; a; B; C # INCOG.: A; b; c; Sup."
640 GOSUB2130
650 GOSUB1970:IFL(1)<=0THENGOSUB1910:GOSUB2160:GOTO 620
660 GOSUB2000:IFLO>7THENGOSUB1910:GOSUB2190:GOTO 620
670 IFA(2)<=0ORA(2)>=PITHENGOSUB1910:GOSUB2170:GOTO 620
680 GOSUB2010:IFLO>7THENGOSUB1910:GOSUB2190:GOTO 620
690 IFA(3)<=0ORA(3)>=PITHENGOSUB1910:GOSUB2170:GOTO 620
700 IFA(2)+A(3)>PYTHENGOSUB1910:GOSUB2180:GOTO620
710 A(1)=PI-A(2)-A(3)
720 GOSUB2020
730 GOSUB2030
740 S=.5*L(1)^2*SIN(A(2))*SIN(A(3))/SIN(A(1))
750 GOSUB1830
760 IFL(1)>LSORL(2)>LSORL(3)>LSORL(4)>LSTHENGOSUB1910:GOSUB2200:GOTO620
770 GOTO1600
780 'tercer caso=====================
790 CLS:PRINT"CCCCCCCCCCC TERCER  CASO CCCCCCCCCCC"
800 PRINT:PRINT"DATOS: a; b; C # INCOG.: A; B; c; Sup."
810 GOSUB2130
820 GOSUB1970
830 GOSUB1980
840 IFL(1)<=0ORL(2)<=0THENGOSUB1910:GOSUB2160:GOTO 790
850 GOSUB2010:IFLO>7THENGOSUB1910:GOSUB2190:GOTO 790
860 IFA(3)<=0ORA(3)>=PITHENGOSUB1910:GOSUB2170:GOTO 790
870 L(3)=SQR(L(1)^2+L(2)^2-2*L(1)*L(2)*COS(A(3)))
880 S=.5*L(1)*L(2)*SIN(A(3)):Y=PI-A(3)
890 IFL(1)<L(2)THEN920
900 X=2*ATN((L(1)-L(2))/(L(1)+L(2))/TAN(A(3)/2))
910 A(1)=(X+Y)/2:A(2)=(Y-X)/2:GOTO 940
920 X=2*ATN((L(2)-L(1))/(L(2)+L(1))/TAN(A(3)/2))
```

```
930 A(1)=(Y-X)/2:A(2)=(Y+X)/2
940 GOSUB2030
950 GOSUB1830
960 IFL(1)>LSORL(2)>LSORL(3)>LSORL(4)>LSTHENGOSUB1910:GOSUB2200:GOTO 790
970 GOTO 1600
980 ' cuarto caso===================
990 CLS:PRINT"CCCCCCCCCCC CUARTO  CASO CCCCCCCCCCC"
1000 PRINT:PRINT"DATOS: a; b; c ≈ INCOG.: A; B; C; Sup."
1010 GOSUB2130
1020 GOSUB1970
1030 GOSUB1980
1040 INPUT"          » Lado c=";L(3)
1050 IFL(1)<=0ORL(2)<=0ORL(3)<=0THENGOSUB1910:GOSUB2160:GOTO 990
1060 IFL(1)+L(2)<=L(3)ORL(1)+L(3)<=L(2)ORL(2)+L(3)<=L(1)THENGOSUB1910:LOCATE2,14
:PRINT"Los lados no conforman un triángulo":GOSUB2060:GOTO 990
1070 P=(L(1)+L(2)+L(3))/2
1080 P1=P-L(1):P2=P-L(2):P3=P-L(3)
1090 R=SQR(P1*P2*P3/P):S=P*R
1100 A(1)=2*ATN(R/P1)
1110 A(2)=2*ATN(R/P2)
1120 A(3)=2*ATN(R/P3)
1130 GOSUB2030
1140 GOSUB1830
1150 IFL(1)>LSORL(2)>LSORL(3)>LSORL(4)>LSTHENGOSUB1910:GOSUB2200:GOTO 990
1160 GOTO 1600
1170 ' quinto caso==================
1180 CLS:PRINT"CCCCCCCCCCC QUINTO  CASO CCCCCCCCCCC"
1190 PRINT: PRINT"DATOS: a; b; A ≈ INCOG.: B; C; c; Sup."
1200 GOSUB2130
1210 PRINT:INPUT" » Lado a";L(1)
1220 INPUT" » Lado b";L(2)
1230 IFL(1)<=0ORL(2)<=0THENGOSUB1910:GOSUB2160:GOTO 1180
1240 INPUT" » Ang. A";A(1):LO=FNLO(A(1)):A(1)=FNR(A(1)):IFLO>7THENGOSUB1910:GOSU
B2190:GOTO 1180
1245 IFA(1)<=0ORA(1)>=PITHENGOSUB1910:GOSUB2170:GOTO1180
1250 C=L(2)*SIN(A(1))
1260 IFA(1)<PI/2THEN1450
1270 IFL(2)>=L(1)THEN1380
1280 LOCATE20,7:PRINT"* UNICA SOLUCION *"
1290 R=L(2)/L(1)*SIN(A(1))
1300 A(2)=ATN(R/SQR(1-R^2))
1310 A(3)=PI-A(1)-A(2)
1320 L(3)=L(1)*SIN(A(3))/SIN(A(1))
1330 GOSUB2030
1340 S=.5*L(1)*L(2)*SIN(A(3))
1350 GOSUB1830
1360 IFL(1)>LSORL(2)>LSORL(3)>LSORL(4)>LSTHENGOSUB1910:GOSUB2200:GOTO 1180
1370 GOTO 1600
1380 RESTORE2260
1390 FORN=1TO17:READH$
1400 LOCATEN+10,15:PRINTH$ ; BEEP
1410 RE=50:GOSUB2050
1420 NEXTN
1430 RE=1500:GOSUB2050
1440 GOTO 1180
1450 IFL(1)>=L(2)THEN1280
1460 IFL(1)>CTHEN1490
1470 IFL(1)=CTHEN1280
1480 GOTO1380
1490 LOCATE24,6:PRINT"Tiene  dos"
1500 LOCATE24,7:PRINT"Soluciones"
1510 LOCATE20,8:PRINT"* 1ra. SOLUCION *":GOTO 1290
1520 PRINT:PRINT"     [2da. SOLUCION -     tipee 2]"
1530 D$=INKEY$
1540 K=22:L=22:GOSUB2110
1550 IFD$=""ORD$<>"2"THENGOSUB2120:GOTO 1530
1560 LOCATE22,8:PRINT"2d":BEEP
1570 A(2)=PI-A(2):A(3)=PI-A(1)-A(2)
1580 PRINT:GOTO 1320
1590 ' rutina de visualización=========
1600 LOCATE0,9:GOSUB2140
1610 FORN=1TO11
1620 PRINT"|
1630 NEXTN
1640 GOSUB2150
1650 V$="&=#######.##  &=###°##'##''"
1660 RESTORE2240:Q=10:FORN=1TO4:READE$,F$
1670 LOCATE1,Q:PRINTUSINGV$;E$;L(N);F$;G(N);M(N);S(N)
1680 Q=Q+2
1690 IFN=3THENPRINT"|————————————————————————————————|"
1700 NEXTN
1710 IFINT(ER+.5)<>0THENPRINTUSING"|                    Error de redondeo = ##''";ERELSE
PRINT:PRINT:GOTO 1730
1720 PRINTUSING"|  S(A)=##.#  S(B)=##.#  S(C)=##.#";S(1);S(2);S(3)
1730 PRINT:PRINTUSING"|    SUPERFICIE = ############.##";S
1740 IFVAL(C$)=5ANDC<L(1)ANDL(1)<L(2)ANDA(2)<PI/2THEN1520
1750 PRINT:PRINT"  ¿Resolverá otro triángulo?    [S/N]"
1760 I$=INKEY$
1770 K=30:L=22:GOSUB2110
1780 IFI$=""THENGOSUB2120:GOTO 1760
1790 IFI$="S"ORI$="s"THENRUN
1800 IFI$="N"ORI$="n"THEN1810ELSE1760
1810 CLS:END
1820 ' subrutina conversión angular====
1830 FORN=1TO4
1840 AS(N)=A(N)*180/PI:G(N)=FIX(AS(N)):Z=(AS(N)-G(N))*60:M(N)=FIX(Z):S(N)=(Z-M(N
))*60
1850 IFS(N)>=59.5THENS(N)=0:M(N)=M(N)+1
1860 IFM(N)=60THENM(N)=0:G(N)=G(N)+1
1870 NEXTN
1880 ER=FIX(S(1)+.5)+FIX(S(2)+.5)+FIX(S(3)+.5)-S(1)-S(2)-S(3)
1890 RETURN
1900 ' ============================
1910 RESTORE2250
1920 FORN=1TO31:READG$
1930 LOCATEN+2,12:PRINTG$
1940 RE=50:GOSUB2050
1950 NEXTN
1960 RETURN
1970 PRINT:INPUT"          » Lado a=";L(1):RETURN
1980 INPUT"          » Lado b=";L(2):RETURN
1990 INPUT"          » Ang. A=";A(1):LO=FNLO(A(1)):A(1)=FNR(A(1)):RETURN
2000 INPUT"          » Ang. B=";A(2):LO=FNLO(A(2)):A(2)=FNR(A(2)):RETURN
2010 INPUT"          » Ang. C=";A(3):LO=FNLO(A(3)):A(3)=FNR(A(3)):RETURN
2020 D=L(1)/SIN(A(1)):L(2)=D*SIN(A(2)):L(3)=D*SIN(A(3)):RETURN
2030 A(4)=A(1)+A(2)+A(3)
2040 L(4)=L(1)+FIX(L(2)*100+.5)/100+FIX(L(3)*100+.5)/100:RETURN
2050 FORA=1TORE:NEXT:RETURN
2060 LOCATE0,19:GOSUB2140
2070 PRINT"| Presione la barra espaciadora para |":PRINT"|          reingresar d
atos          |"
2080 GOSUB2150
2090 J$=INKEY$:IFJ$=""THEN2090
2100 IFJ$=" "THENRETURNELSE2090
2110 LOCATEK,L:PRINT"=>":RE=200:GOSUB2050:RETURN
2120 LOCATEK,L:PRINT"=▶":RE=200:GOSUB2050:RETURN
2130 PRINT:PRINT"CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC":RETURN
2140 PRINT"XWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWY":RETURN
2150 PRINT"ZWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW;":RETURN
2160 PRINT:PRINT"  Ingrese lados no nulos y positivos":GOSUB2060:RETURN
2170 PRINT:PRINT"Ingrese áng no nulos;positivos y < 180°":GOSUB2060:RETURN
2180 PRINT:PRINT" La suma de los áng. debe ser < 180°":GOSUB2060:RETURN
2190 PRINT:PRINT"Ingrese áng. bajo la forma GRADOS.MMSS":GOSUB2060:RETURN
2200 PRINT:PRINT"El formato de impresión es desbordado":PRINT"      Ingrese magn
itudes menores":GOSUB2060:RETURN
2210 ' datas=========================
2220 DATA53,75,1,203,75,2,53,145,3,128,145,4,203,145,5,52,92,a,205,92,a,52,162,a
,127,162,a,202,162,a,115,50,DATOS,53,46,A,75,90,B,225,90,B,185,90,C,37,135,b,30,
162,C,215,135,c,185,160,C,115,135,b,142,135,c
2230 DATA40,65,b,68,65,c,30,90,C,203,46,A,70,162,B,190,65,b,215,65,c,53,116,A,12
8,116,A,203,116,A,150,162,B,225,162,B,110,162,C,190,135,b,70,135,c,
2240 DATALado a,Ang.A,Lado b,Ang.B,Lado c,Ang.C,ΣLados,Ang.Σ
2250 DATA ¡,C,O,N,T,R,O,L,E,,L,O,S,,D,A,T,O,S,,I,N,G,R,E,S,A,D,O,S,!
2260 DATAN,O,,T,I,E,N,E,,S,O,L,U,C,I,O,N
```

# MSX-TEST

CONCURSO MENSUAL

# PREMIO



*Un soft a eleccion entre IDEA BASE, IDEA TEXT, o BASIC TUTOR.*

**Auspiciado por TELEMATICA S.A. fabricante en Argentina de las computadoras personales TALENT MSX**

Para participar en este certamen deben señalar cual es la información correcta que presenta cada ítem. Junto con las respuestas deben remitir los datos en el correspondiente cupón. Los premios se entregarán en la administración de la editorial. Los que no puedan concurrir pueden solicitar el envio pagando el franqueo contrareembolso. Los premios podrán ser reclamados dentro de los 120 días después de haber sido anunciados.

**CIERRE 1º DE MARZO DE 1988**

**1) En la Talent MSX ¿Cómo se denomina la región de memoria comprendida entre las direcciones FDGAH y FFCGH?**
a) núcleo de estado
b) área de hooks
c) memoria primitiva

**2) ¿Qué es BASICA?**
a) El lenguaje BASIC de IBM
b) Un procesador de texto para MSX
C) El lenguaje BASIC de las MSX2

**3) ¿Para qué sirve la función LEN?**
a) Hace funcionar la computadora a 0,5 Mhz.
b) Ordena alfabeticamente las variables alfanuméricas.
c) Indica cuantos caracteres contiene una expresión alfanumérica.

**4) Cuántos modos de pantalla tiene la MSX2?**
a) 9
b) 12
c) 16

**5) ¿Qué significa el término "benchmark"?.**
a) Marca de nivel
b) Prueba de escritorio
c) Estudio de Flujo.

Nombre y apellido: ____________________

Dirección: ____________________

Documento: ____________ Edad: ______ Máquina: ____________

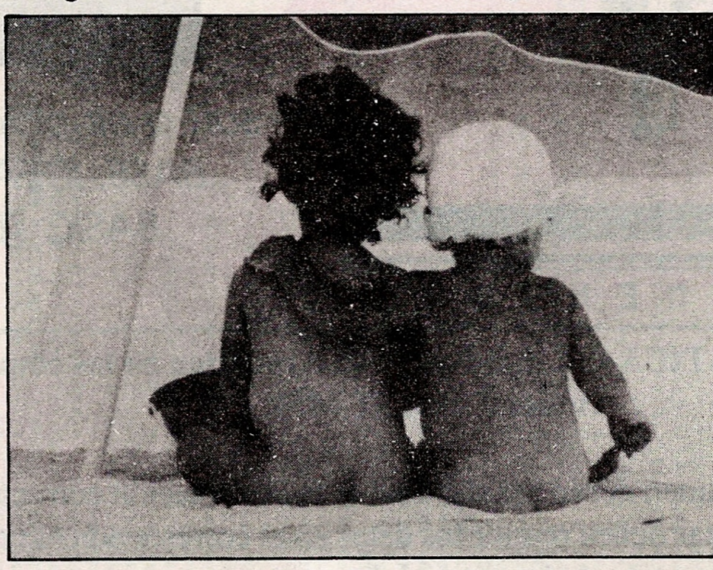
Qué es lo que más me gusta de la revista: ____________________

Que le agregaría: ____________________

Que es lo que no me gusta: ____________________

# DESNUDANDO PROGRAMAS

***¿Cómo se estructura en la memoria RAM un programa? ¿Dónde comienza y cómo se puede generar SOFTWARE que se modifique a sí mismo? Lo veremos en esta nota.***



Para comenzar nos referiremos al programa de la figura 1, que nos permitirá observar cómo un programa (conjunto de letras, números y símbolos) puede guardarse en "cajas" (celdas de memoria)que aceptan únicamente números comprendidos entre 0 y 255.

En la primera columna de la citada figura se presenta un número identificatorio de la celda de memoria, llamado dirección. En la segunda aparece el contenido de la celda antes identificada, cuyos valores están comprendidos entre 0 y 255. La tercera columna muestra lo que significa la segunda en el código ASCII.

Al observar el programa, vemos que este se compone de líneas que están identificadas con un número (10, 20 y 30) y cada línea tiene una o varias palabras claves (REM, PRINT, FOR, etcétera) y un contenido (ABC, "DIREC",etcétera).

Si analizamos las direcciónes 32769 y 32770 veremos que al multiplicar el contenido de la segunda por 256 y sumarle la primera nos da el valor donde comienza la próxima línea cuyo número es 20 (ver comentarios de ejecución de programa 1).

Las dos direcciones siguientes entregan el valor de la línea actual de la misma forma que antes, el contenido de la segunda se multiplica por 256 y, sumándole el de la primera, nos da 10. La dirección 32773 contiene la palabra clave REM y nos damos cuenta de ello porque el observar la tercera columna de las direcciónes 32774, 32775 y 32776 aparecen allí las letras A,B,C, que son el contenido del REM. El fin de línea está dado en la 32777 con un 0. Entonces todo vuelve a repetirse. Los dos primeros dan el comienzo de la próxima línea, los dos siguientes el número de la línea analizada, la palabra clave, en este caso PRINT, y, por último su contenido. Así continúa hasta el fin del programa. Ya que han sido presentadas dos herramientas nos dedicaremos ahora a los programas "extraños". El caso más simple es el de la figura 3, llamado desalineado, pues luego de ejecutarlo, al listarlo, aparece en forma muy poco presentable. El corazón de este programa es la primera línea, que deposita un número que guarda la organización del programa para la presentación. Al modificarla, todo aparece confuso menos la primera línea. Veamos la figura 1 y descubriremos que no sabemos dónde comienza la línea 20, ni la 30, y por lo tan-

## FIGURA 1 "RAYOS X"

| | COMENTARIOS SOBRE EL PROGRAMA |
|---|---|
| 10 REMABC | LINEA DE COMENTARIO |
| 20 "PRINT" DIREC"; "CONTE"; "CAR" | IMPRIME CARTELES PARA VISUALIZACION |
| 30 FOR I=32769 TO 32786; PRINT I; PEEK (I); CHR$ (PEEK(I)): NEXT I | IMPRIME DIRECCION, CONTENIDO Y CARACTER CORRESPONDIENTE AL CONTENIDO |
| RUN | |

**COMENTARIOS SOBRE LA EJECUCION**

| DIREC | CONTE | CAR | IMPRIME CARTELES. |
|---|---|---|---|
| 32769 | 10 | | EL CARACTER ES UNA LINEA EN BLANCO. |
| 32770 | 128 | C | 128X256+10=32778 DIRECCION DE COMIENZO DE PROXIMA LINEA. |
| 32771 | 10 | | 0X256+10=10 NUMERO DE LINEA |
| 32772 | 0 | | |
| 32773 | 143 | A | NUMERO QUE CORRESPONDE AL REM. |
| 32774 | 65 | A | |
| 32775 | 66 | B | CONTENIDO DEL REM. |
| 32776 | 67 | C | |
| 32777 | 0 | | FIN DE LINEA |
| 32778 | 41 | ) | 128X256+41=32811 AQUI COMIENZA LA LINEA SIGUIENTE. |
| 32779 | 128 | C | |
| 32780 | 20 | | 0X256+20=20 NUMERO DE LINEA. |
| 32781 | 0 | | |
| 32782 | 145 | 32 | NUMERO QUE REPRESENTA AL PRINT. |
| 32783 | 34 | " | COMILLAS QUE DAN COMIENZO AL CONTENIDO DEL PRINT. |
| 32784 | 32 | | |
| 32785 | 68 | D | |
| 32786 | 73 | I | |

## FIGURA 2 "DETECTIVE"

| Programa | COMENTARIO SOBRE EL PROGRAMA |
|---|---|
| 20 FOR I= 32500 TO 65532 | PARA IR VARIANDO ENTRE 32500 Y 65532. |
| 30 IF PEEK(I)=PEEK(I+1)-1 AND PEEK(I)=PEEK(I+2)-2 AND PEEK(I)= PEEK(I+3)-3 THEN 60 | SI EL CONTENIDO DE LA DIRECCION ES IGUAL AL DE LA SIGUIENTE MENOS UNO, TAMBIEN A LA POSTERIOR, MENOS DOS Y TAMBIEN A OTRA MAS POSTERIOR ENTONCES VAYA A 60. |
| 40 NEXT I | BUSQUE EL PROXIMO I. |
| 50 PRINT"NO HAY NADA CONSECUTIVO":END | AVISA QUE NO ENCONTRO NADA. |
| 60 PRINT" HAY ALGO CONSECUTIVO EN";I | AVISA LA DIRECCION ENCONTRADA. |
| 10 REMABCD | PRIMER CASO: COLOCAMOS PISTA. |
| RUN | EJECUTAMOS EL PROGRAMA. |
| HAY ALGO CONSECUTIVO EN 32774 | INFORMA. |
| 10 | BORRO PRIMERA PISTA. |
| SAVE"ABCD" | SEGUNDO CASO: COLOCAMOS LA PISTA. |
| RUN | CORREMOS EL PROGRAMA. |
| HAY ALGO CONSECUTIVO EN 63590 | INFORMA. |
| SAVE"MSX" | BORRO LA SEGUNDA PISTA. |
| KEY 1, "ABCD" | TERCER CASO: UBICAMOS LA PISTA. |
| HAY ALGO CONSECUTIVO EN 63615 | INFORMA. |

## FIGURA 3 "DESALINEADO"

| Programa | COMENTARIOS SOBRE EL PROGRAMA |
|---|---|
| 10 POKE 32769,255 | ANULA LA ORGANIZACION DEL PROGRAMA PARA LA PRESENTACION CON LIST. |
| 20 INPUT A$ | VERIFICACION DE LA EJECUCION |
| 30 PRINT A$ | SI INGRESA NUEVAMENTE LINEA 10 SE RESETEA LA COMPUTADORA |

## FIGURA 4 "CARGADOR"

| Programa | COMENTARIO SOBRE EL PROGRAMA |
|---|---|
| 10 REM"PEPE" | LINEA QUE CARGARA EL PROGRAMA; POR AHORA ES INUTIL. |
| 20 POKE 32773,181 | COLOCA EN LUGAR DEL REM UN LOAD. |
| 30 GOTO 10 | VA A CARGAR EL PROGRAMA LLAMADO PEPE QUE ESTA EN DISQUETE. |

to la presentación del listado será muy confusa, salvo en caso de que tengamos la disquetera conectada. Otro programa "extraño" es el de la figura 4. Detrás de un programa aparentemente inofensivo, que no sería nada más que un ir y venir entre la primera y última línea, se esconde un cargador de programas.

El secreto está nuevamente en el POKE que coloca, en lugar de la sentencia REM, la instrucción LOAD.

Existen otros programas que son más cambiantes aún, como es el caso del de la figura 5, que consiste en alternar INPUT y PRINT en una sola dirección a medida que corre el programa. La explicación es relativamente sencilla: en la primera línea se establece el valor inicial a X (el número que le corresponde al PRINT), luego se realiza un INPUT y más adelante se le asigna a la variable Y el contenido de la dirección 32782 (que corresponde al número que representa la sentencia INPUT, que acabamos de ejecutar). Después se hace un cambio de valores con la sentencia SWAP.

Es decir que lo que valía X pasa a valer Y y viceversa, y por último se deposita el valor de X en la dirección donde se encontraba el INPUT. El GOTO 20 permite que no pare el programa, después de ingresar un valor y de imprimirlo, para volver a pedir otro.

Hasta ahora cambiamos sentencias y números que organizan el programa, desde aquí en adelante cambiaremos de sentencias y lugares elegidos de memoria.

El programa de la figura 6 se basa en la variación del número asignado a una variable Z. Ese valor se incrementa a medida que se corre el programa es decir, cuando se corre por primera vez Z vale cero. Al encontrar el POKE

## FIGURA 5 "DOBLE PERSONALIDAD"

COMENTARIOS SOBRE EL PROGRAMA

| | |
|---|---|
| 10 X=145 | SE ASIGNA A X EL NUMERO QUE REPRESENTA AL PRINT |
| 20 INPUT "VAMOS"; A$ | SE INGRESA UNA VARIABLE. |
| 30 Y=PEEK (32782) | SE ASIGNA A Y EL NUMERO QUE REPRESENTA AL INPUT |
| 40 SWAP X,Y | CAMBIA LOS CONTENIDOS DE LAS VARIABLES. |
| 50 POKE 32782,Y | SE DEPOSITA EL PRINT EN LUGAR DEL INPUT. |
| 60 GOTO 20 | VA A IMPRIMIR A$. |

EL PROGRAMA ES UN POCO OBSECUENTE

## FIGURA 6 "MORTAL"

COMENTARIO SOBRE EL PROGRAMA

| | |
|---|---|
| 10 Z=0 | ASIGNA A Z EL VALOR CERO. |
| 20 Y=32775 | ASIGNA A Y LA DIRECCION DONDE ESTA EL "CERO" |
| 30 POKE Y, PEEK(Y)+1 | COLOCA EL VALOR ANTERIOR PERO INCREMENTADO. |
| 40 PRINT" LE QUEDAN"; 3-Z; "OPORTUNIDADES" | INFORMA |
| 50 IFZ=3 THEN POKE32808,143 | SI Z LLEGA A TRES, CAMBIA EL PRINT DE LA 40 POR UN REM. |

## FIGURA 7 "VIRGEN"

COMENTARIOS SOBRE EL PROGRAMA

| | |
|---|---|
| 10 POKE 63590,49 | COLOCA UN UNO EN CODIGO ASCII EN LA DIRECCION DE ALMACENAMIENTO DE LOS NOMBRES DE PROGRAMAS QUE ENTRAN O SALEN DE LA COMPUTADORA, ESTOS NOMBRES NO PUEDEN EMPEZAR CON UN NUMERO, ES DECIR, JAMAS HABRIA UN UNO. |
| 20 POKE 32773,143 | COLOCA UN REM EN LUGAR DEL POKE ANTERIOR. |
| 30 IF PEEK(63590)=49 THEN 40 ELSE STOP | SI LA MODIFICACION CONTINUA, EL PROGRAMA SIGUE; DE LO CONTRARIO SE DETIENE. |
| 40 PRINT"ESTE PROGRAMA ANDA" | INFORMA QUE ANDA. |

## FIGURA 8 "TIMIDO"

COMENTARIO SOBRE EL PROGRAMA

| | |
|---|---|
| 10 INPUT"INGRESE ALGO";A$ | INGRESA A$ |
| 20 PRINT"USTED INGRESO ";A$ | IMPRIME LO INGRESADO (NO SE OLVIDE DEL ESPACIO) |
| 30 POKE 32842,255 | COLOCA UN 255 EN LUGAR DE LA Z. |
| 40 DEFFNA(X)=Z4 | SE LOGRA UN CLS AL LISTAR TODO O LA LINEA 40 NADA MAS. |

coloca un valor en la dirección donde se encuentra el supuesto cero; ese valor es el mismo leído en aquella dirección, pero incrementado en uno; es decir, si lee en la misma dirección donde se encuentra el cero un 49, coloca 50. Esto significa que cuando corra nuevamente el programa a Z se le asignará uno, y luego, por medio del POKE, la primera línea pasará a asignar a Z el valor dos, y así sucesivamente. Por supuesto todo tiene su límite y, si el número es 3, existen dos posibilidades. La primera, si tiene disquetera, es destruir el programa y grabarlo. Así no podrá volver a utilizarse nunca. La segunda es arruinar el programa que se está corriendo. En el caso de la figura 6, se saca el PRINT y se coloca un REM. Debemos aclarar que la forma de guardar un número que se va a asignar no es tan simple como se presentó, pero vale como modelo didáctico.

La figura 7 muestra un programa realmente interesante ya que conjuga dos conceptos: la modificación de un sector de la memoria y la modificación del programa.

El primer cambio se realiza sobre la zona donde se almacena el nombre de los programas que ingresan a la computadora o egresan de ella. La dirección de memoria para la modificación se obtiene del programa de la figura 2. La segunda modificación se realiza sobre la instrucción que hace la primera, por lo tanto esta última se lleva a cabo una sola vez (cuando corre el programa en la primera oportunidad). Si esta se detiene, la zona de memoria guarda el nombre del programa y no la modificación hecha. Si consideramos que el programa grabado y el programa en la memoria no son capaces de restituir la modificación necesaria para andar debido a la segunda modificación, entonces ninguno de los dos programas sirven. Este último programa es instructivo, ya que si el programa grabado corre, no puede ser grabado, entonces tenemos un camino más por el cual podríamos evitar la copia.

La figura 8 nos presenta el programa llamado "tímido" porque luego de copiarlo y correrlo, se produce un efecto de borrado al intentar listarlo como si el programa no quisiera presentarse ante nosotros. El efecto se consigue al tener la sentencia de definición de función seguida de una serie de números; el que agregamos es el 255. Claro que experimentando podremos obtener otros efectos más.

**Miguel Luis Kopec**

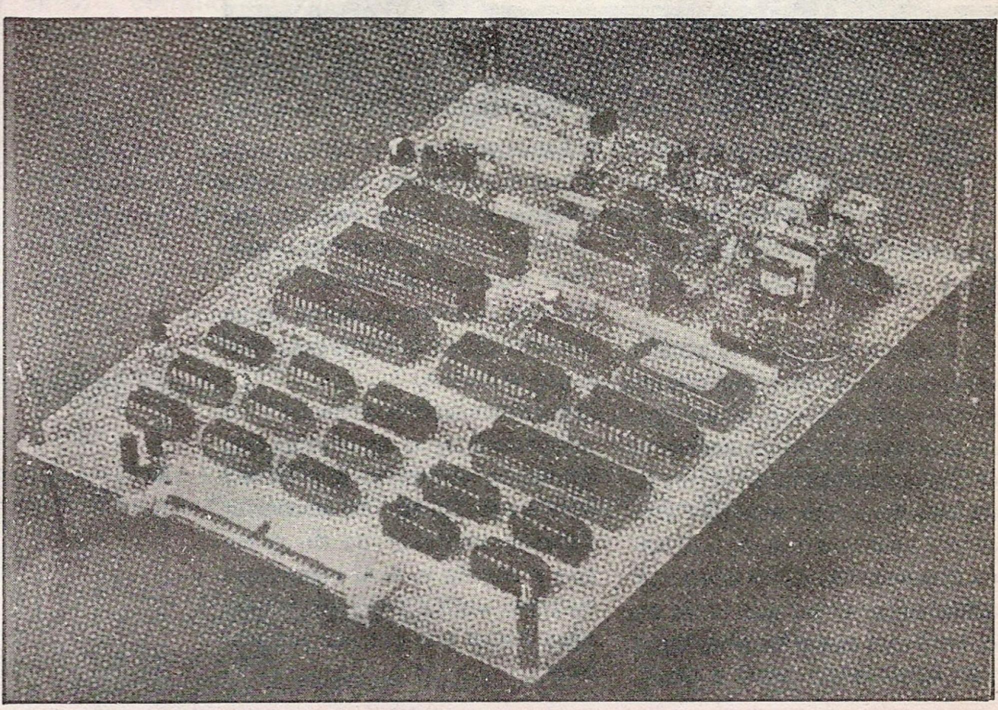
# FACSIMILES EN MSX

***La MSX es una máquina multipropósito que, según los periféricos que se conecten, puede servir como procesador de textos o terminal para comunicaciones. Pronto logrará hacer un facsímil.***

Su nombre es MSX-FAX. Es un cartucho que hay que insertar en el slot y que nos permite hacer facsímiles.

Su manejo es igual al del cartucho de comunicaciones. Los envíos vía FAX normalmente requieren una hoja tipeada y/o manuscrita para insertar dentro de la máquina y luego marcar el número de teléfono-destino. Con el MSX-FAX este procedimiento no es necesario, ya que lo que fue tipeado en el MSX-Write es enviado directamente, es decir, que nos evitamos el proceso de impresión. También se pueden memorizar números de teléfonos-destino y simplificar la comunicación apretando una sola tecla.

El MSX-FAX permite enviar una hoja tamaño A4 en aproximadamente 30 a 60 segundos , variando el tiempo según el espacio en blanco que posee cada hoja.



**Prototipo del MSX-FAX, actualmente en desarrollo.**

## EL SOFT INCORPORADO

Cuando se inserta el cartucho aparece en pantalla el menú principal del MSX-Write, con el agregado del FAX.

Tenemos a nuestra disposición índices telefónicos, detalles de envío, "headers" o encabezamientos (primera línea que sirve como distintivo de cada mensaje).

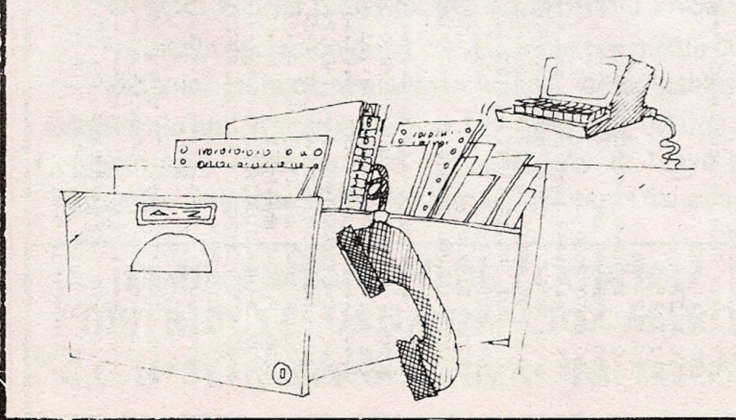
Indice telefónico: puede memorizar hasta 40 registros por vez, y por supuesto queda grabado aún después del apagado de la consola. Asimismo, como también es posible guardar los datos en discos, no hay límite para estos.

Se pueden incluir nombre de empresa, dirección, contactos, número de facsímil, números de teléfono y código. El número del destinatario es discado automáticamente, aunque obviamente también es posible hacerlo en forma manual.

Headers: lo más usual es que en la primera línea del FAX que se envía se coloque un informe codificado del remitente. Por ejemplo, se suele incluir el nombre de la empresa, hora, firmante, etcétera. En MSX-FAX se incluye una función de edición que permite el envío de, por ejemplo, el logotipo de la empresa (también grabado en la RAM) . Detalle del envío: es habitual incluir como primer mensaje los datos del remitente, del destinatario y la referencia. Con el MSX-FAX es posible especificar el formato de los mensajes, según los datos que ya indicamos en el índice telefónico .

El MSX-FAX está aún en desarrollo. Creemos que pronto podremos dar más detalles y fecha de salida al mercado mundial. Mientras tanto, esperemos.

## PARA COMUNICARSE




**Para mejorar nuestro servicio, a los teléfonos de Editorial Proedi hemos incorporado el sistema de Radio Llamada. Pueden marcar:**

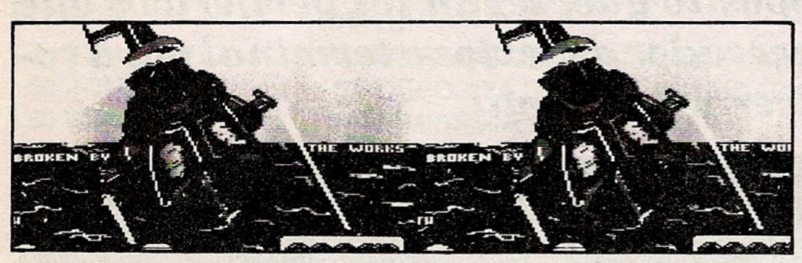
**311-0056 y 312-6383 (cadena de 19 líneas), mencionando el**

**Código 5941**

**y de esa forma podrán dejar el mensaje, que será transmitido de inmediato a nuestras oficinas. Contestaremos a la brevedad (por la vía que corresponda) todas las llamadas.**

## CERVEUX

# HELICOPTERO




Este programa fue desarrollado por los alumnos del curso de Basic Jóvenes dictado entre septiembre y octubre de este año en el Club.

Se trata de un juego de acción en el que tenemos que disparar a los helicópteros enemigos desde nuestro cañón antiaéreo. Si disparamos cuando hay una bala en el aire, esta desaparece y sale una nueva. Eso hace que sea más difícil acertar, y además tenemos que tener en cuenta que nuestros enemigos van acelerando poco a poco.

Comenzamos con 1000 puntos y nos dan 100 por máquina derribada, pero cada disparo nos resta 75, así que tenemos que afinar mucho la puntería.

Se juega con los cursores hacia los costados y la barra para disparar el cañón.

Los chicos (todos entre 14 y 16 años) dedicaron las últimas dos semanas del curso a producir este juego. En él van a encontrar detalles interesantes de gráficos y sonido y, si lo investigan un poco, varios "truquitos" útiles para hacer sus propios programas.

```
1 ' +--------------------+
2 ' |   * Helicoptero *  |
3 ' +--------------------+
4 ' | por: Basic Jovenes |
5 ' |     9/10 del '87   |
6 ' +--------------------+
7 ' |                    |
8 ' |  Cristina   Farias |
9 ' |                    |
10 '|  Beatriz     Primo |
11 '|                    |
12 '|  Esteban Broitman  |
13 '|                    |
14 '|  Christian Chiera  |
15 '|                    |
16 '|  Martin      Moya  |
17 '|                    |
18 '|                    |
19 '+--------------------+
20 COLOR 15,1,1:SCREEN 2,2,,,0:N=3:PC=3:SOUND 7,55:SOUND 6,20:SOUND13,10:SOUND11
,200
30 ON SPRITE GOSUB 6000:SPRITE ON:OPEN"GRP:"AS#1:ON STRIG GOSUB 5000
40 CX(1)=110:CY(1)=137:FX(1)=103:FY(1)=130
50  CX(2)=116:CY(2)=134:FX(2)=112:FY(2)=125
60 CX(3)=123:CY(3)=133:FX(3)=123:FY(3)=120
70  CX(4)=129:CY(4)=134:FX(4)=133:FY(4)=125
80  CX(5)=135:CY(5)=137:FX(5)=142:FY(5)=130
90 R(1)=.82:R(2)=.4:R(3)=0:R(4)=-.4:R(5)=-.82
100 S(1)=0:S(2)=60:S(3)=123:S(4)=183:S(5)=255
110 GOSUB 10000
120 KEYOFF:COLOR 15,1,1:CLS
130 LINE(0,173)-(255,191),9,BF
140 LINE(108,174)-(138,150),14,BF
150 CIRCLE(123,150),16,14,-.1,-3.14,1.33:PAINT(130,148),14
160 LINE(CX(3),CY(3))-(FX(3),FY(3)),14
165 PSET(10,180),9:COLOR1:PRINT#1,"PUNTOS: 1000":SC=1000:STRIG(0)ON
166 PSET(170,180),9:PRINT#1,"NIVEL: 1"
170 SOUND8,16
179 '**** Pone Helicopteros *****
180 X(1)=255
190 X(2)=0
210 Y(1)=1
220 Y(2)=20
240 ON INTERVAL =5 GOSUB 1000
250 INTERVAL ON
260 GOTO 2000
999 ' ******** Lazo principal ********
1000 INTERVAL OFF:SPRITE OFF:STRIG(0) OFF
1005 PUT SPRITE 0,(X(1),Y(1)),15,1
1010 PUT SPRITE 1,(X(2),Y(2)),15,2
1020 X(1)=X(1)-N
1025 X(2)=X(2)+N
1030 INTERVAL ON:SPRITE ON:STRIG(0) ON
1040 RETURN
1999 ' **** Bala *****************
2000 IF F THEN YB=YB-3:IF YB<0 THEN F=0 :PUT SPRITE 3,(255,191),0,3:ELSE PUT SPR
ITE 3,((YB*R(DR))+S(DR),YB),15,6
2999 '***** MUEVE CAÑON *************
3000 SK=STICK(0):IF SK=0 THEN 2000
3010 LINE(CX(PC),CY(PC))-(FX(PC),FY(PC)),1
3020 IF SK=3 AND PC<5 THEN PC=PC+1
3030 IF SK=7 AND PC>1 THEN PC=PC-1
3999 '***** Dibuja Cañon ************
4000 LINE(CX(PC),CY(PC))-(FX(PC),FY(PC)),14
4010 GOTO 2000
4999 '***** Disparo ****************
5000 SC=SC-75
5005 LINE(115,191)-(70,180),9,BF:COLOR1:PRINT#1,SC:IF SC<0 THEN 20000
5010 F=1:YB=FY(PC):DR=PC:RETURN
5999 '***** Colision **************
6000 SPRITE OFF:INTERVAL OFF:STRIG(0)OFF
6010 IF YB<20 THEN HM=1 ELSE HM=2
6020 F=0:PUT SPRITE 3,(255,191),0,3:D=3:FOR Q=15 TO 1 STEP -.3:SOUND 8,Q:COLOR ,
,Q
6030 D=D+.02:PUT SPRITE HM-1,(X(HM),Y(HM)),6,D:NEXT
6040 X(HM)=0
6050 SC=SC+100:N=N+.2:SOUND11,200-N*15
6060 LINE(115,191)-(70,180),9,BF:COLOR1:PRINT#1,SC
6070 LINE(256,191)-(220,180),9,BF:PRINT#1,INT(N-2)
6900 SPRITE ON:INTERVAL ON:STRIG(0)ON:SOUND8,16:RETURN
9999 '**** SPRITES ****************
10000 FOR Q=1 TO 5
10010     FOR W=1 TO 32:READ A$
10020         SP$=SP$+CHR$(VAL("&H"+A$))
10030 NEXT W:SPRITE$(Q)=SP$:SP$="":NEXT Q
10040 SPRITE$(6)="LL"+STRING$(6,CHR$(0))
10999 RETURN
11000 '*******************************
11010 DATA 00,FF,02,1F,24,44,47,FF,FF,7F,21,FF,00,00,00,00,00,F8,00,C0,62,3F,FA,
E0,C0,80,00,C0,00,00,00,00
11020 DATA 00,1F,00,03,46,FC,5F,07,03,01,00,03,00,00,00,00,00,FF,40,F8,24,22,E2,
FF,FF,FE,84,FF,00,00,00,00
11030 DATA 22,00,0A,20,85,22,0A,25,1F,07,29,42,04,48,02,21,80,2A,40,A4,D0,96,6C,
F8,F0,F8,EC,D1,44,A1,12,00
11040 DATA 00,00,00,04,01,02,08,05,21,0A,00,05,08,02,09,10,00,00,00,50,10,44,28,
40,B0,A0,40,30,88,24,40,00
11050 DATA 00,00,00,00,00,01,04,02,02,08,02,00,0A,10,22,08,00,00,00,00,00,50,00,
A0,08,D0,00,28,40,10,84,12
20000 INTERVAL OFF:SPRITE OFF:STRIG(0)OFF
20010 COLOR 15:SOUND8,0:PRESET (20,50):PRINT #1,"G A M E  O V E R"
20020 PRESET (18,70):PRINT #1,"Jugás de nuevo (s/n)?"
20030 K$=INKEY$
20040 IF K$="S" OR K$="s" THEN RUN
20045 IF K$="N" OR K$="n" THEN DEFUSR=1:PRINTUSR(1)
20050 GOTO 20030
```

## CURSO DE OPERACION DE DBASE II



El DBase es uno de los utilitarios más conocidos, y actualmente su uso está muy difundido por los buenos resultados que produce su aplicación en computadoras MSX.

Para el no iniciado en la programación o en la operación de sistemas es muy sencillo dominar esta herramienta de manejo de archivos.

Planteamos el proceso de aprendizaje de DBase II en tres etapas independientes: Operación (que es la que detallamos en esta nota), Programación y Taller de aplicaciones (que desarrollaremos más adelante).

Esta primera parte incluye los conceptos teóricos básicos sobre archivos y sus componentes: Campos y registros; los modos de apertura Input, Output e Input/Output y las organizaciones y accesos: Secuencial, Relativo, Indexado, Dinámico, Random.

El segundo paso es revisar algunos inconvenientes en las instalaciones tradicionales, y las formas de solucionarlos. Aquí surge el concepto de Base de Datos, qué es un Administrador de Bases de Datos y cómo se crean las bases.

Allí empieza el trabajo sobre DBase II: crear un archivo, ver su contenido, abrirlo y cerrarlo; luego están los comandos de edición y el uso de expresiones para ampliar las posibilidades de los comandos.

Habiendo practicado mucho todo esto empezamos a acceder a la base de datos por comandos o en forma interactiva y a trabajar con datos en memoria.

Pasada la mitad del curso, ya tenemos elementos como para explorar posibilidades solos, mientras vamos aprendiendo nuevos comandos y el uso de instrucciones lógicas, las formas de modificar la estructura de un archivo, e incluso el traslado de archivos a otros lenguajes. Para finalizar quedan los métodos de modificación y actualización de datos, organización y ordenamiento y búsquedas rápidas.

El curso es de 16 horas cátedra divididas en 8 clases, dos veces por semana.

Como siempre, para cualquier consulta acerca de este o cualquier otro tema, estamos a su disposición en: CLUB DE USUARIOS MSX Av.Córdoba 650 (1054) Capital Federal o en los teléfonos: 392-5328/ 7611/ 8043 8051/ 8251/ 8478/ 8536/ 9515

## VENTANA

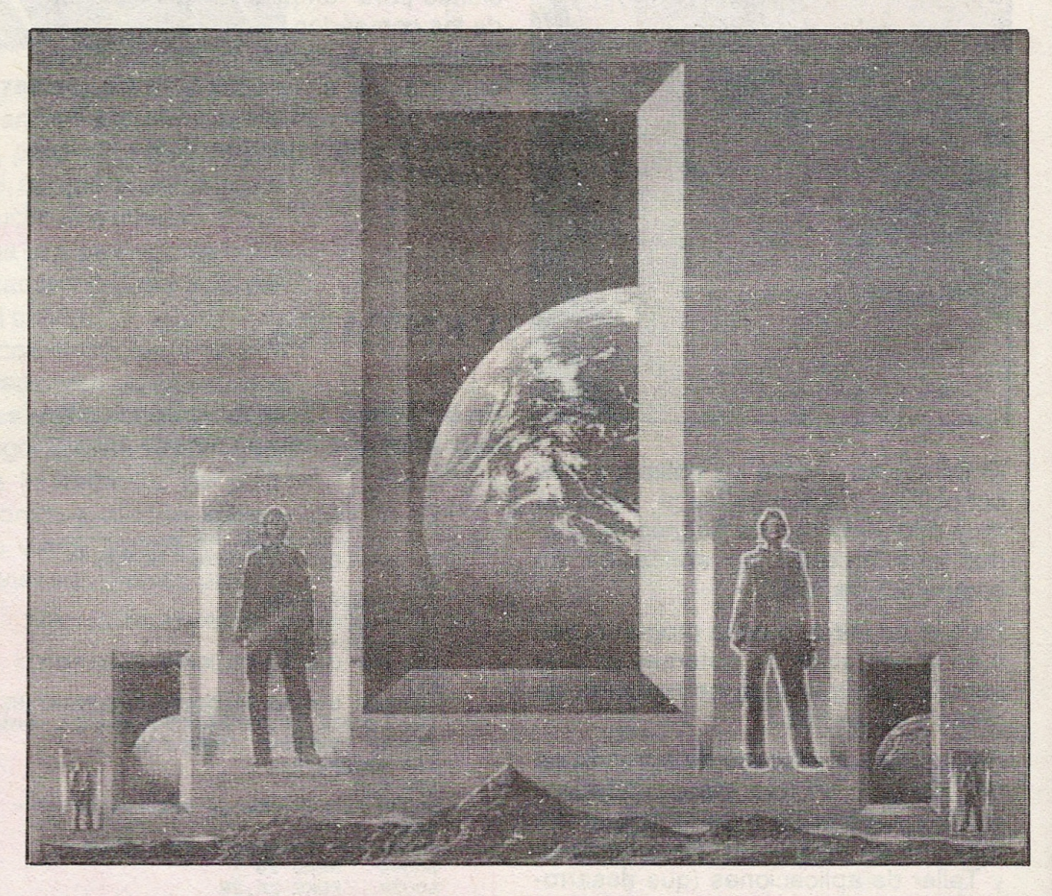
En el número 17 se deslizó un involuntario error que habrán notado aquellos que quisieron ejecutar el programa "ventana". Al mismo le falta la línea de datos. Aquí les ofrecemos el programa completo pidiéndoles disculpas por el mencionado error.

```
10 '-------- VENTANA -----------
20 '---- por M.Salias ----------
30 '--- Club de Usuarios MSX --
40 '
100 GOSUB 60030
110 FOR Q=100 TO 250:PRINT Q;:NEXT
120 A=USR1(0)
130 FOR Q=5 TO 9:LOCATE 5,Q:PRINT"!
EXT
135 FOR Q=5 TO 9:LOCATE 0,Q:PRINT:NEX"
140 LOCATE 10,7:PRINT"Prueba de ventana ..."
150 IF INKEY$="" THEN 150
160 A=USR2(0)
170 END
60000 '------- Carga del código ----
60010 '
60030 FOR ZA=&HD000 TO &HD020
60040   READ ZB
60050   POKE ZA,ZB
60060 NEXT
60070 DEFUSR1=&HD000
60080 DEFUSR2=&HD013
60090 RETURN
60100 DATA 58,175,252,254,0,192,33,205,0,17,32,208,1,230,0,205.89
,0,201,33,32,208,17,205,0,1,230,0,205,92,0,201,0
```

# ASSEMBLER , BYTES Y OTRAS YERBAS

***Presentamos un glosario de términos referidos al Assembler para ayudar al que recién se inicia en la computación. La nota pretende ser una introducción al mundo del código de máquina.***

Podemos observar día a día que, en revistas de nuestro medio y del extranjero, se publican programas de más calidad, que se depuran técnicas de programación, se usan rutinas más compactas y eficientes, se mejora la estética y calidad, etcétera.
Pero así como la belleza y la elaboración tienden a hacer de un programa un buen producto , existe un punto que también debe ser tomado en cuenta : LA VELOCIDAD DE EJECUCION. Este es nuestro fantasma, el que malogra hasta el programa más perfecto, haciéndolo lento en las partes que más se complica, fastidiándonos en la espera de una hermosa pantalla de presentación o aburriéndonos con los desplazamientos interminables de las figuras.
El BASIC es un lenguaje muy flexible y con él logramos hacer programas que en otro idioma son muy complicados de elaborar. Pero es un lenguaje intérprete (es decir, la computadora lo interpreta y ejecuta línea a línea) y se torna demasiado lento. Para evitar este problema se suele recurrir a un lenguaje que, si bien no es tan fácil de emplear, tiene como ventaja la velocidad de ejecución y la potencia.
La velocidad está dada por la máxima que puede dar el microprocesador (el corazón de la computadora) que en el caso de la MSX es de 3.579 Mhz , es decir 3,579 millones de veces más rápida que un segundo. En cambio, la potencia está dada porque éste es el lenguaje interno del sistema, es decir, es el lenguaje que usa el microprocesador para realizar todas las tareas (controlar los circuitos que leen teclados, armar las pantallas, emitir sonidos, grabar datos, etcétera). El BASIC mismo está basado en ese conjunto de códigos y gracias a ellos puede leer, interpretar y ejecutar cualquier instrucción que le escribamos .
El Assembler o ensamblador es un conjunto de códigos que le permiten a un microprocesador (un conjunto de circuitos que realizan múltiples funciones o procesos) ejecutar una serie de operaciones que el fabricante ha grabado en él con antelación. La cantidad de códigos y la variedad determinan, en cierta medida, la "potencia" del micro; en el caso de nuestra computadora dispone de una variedad mayor que150 y una cantidad que supera los 700 .



## ZONAS

A su vez, así como nosotros guardamos nuestros programas en memoria, el micro usa parte de ella para varias operaciones, a saber:
1) Zona con programa de manejo de la computadora
2) Zona de trabajo
3) Zona de almacenaje
4) Zona de control
1 :Si nosotros, cuando realizamos un programa y antes de ejecutarlo, ponemos la instrucción TRON, observaremos que aparecerán en la pantalla los números de línea que se están ejecutando. O sea que nuestra máquina lee y ejecuta constantemente. También el micro lo hace, con la variante de que las líneas son direcciones de memoria y el programa es el que está alojado en ROM. Este programa contiene el MSX BASIC. Se trata de 32768 códigos que permiten al micro leer la tecla que presionamos, aguardar a que terminemos de teclear, y, cuando le demos "RETURN", ejecutar, mostrar en pantalla, grabar, cargar, emitir un sonido o poner un mensaje de error según corresponda. Todo esto en tiempo real, en un instante. A este programa se lo llama SISTEMA OPERATIVO y además de permitir que escribamos en BASIC informándonos de los errores cometidos, pone caracteres en pantalla, maneja el grabador de casetes o disquetera. Además inicializa limpiando la memoria y mostrando el mensaje de presentación, reconociendo un cartucho cuando está pues-

to y muchas cosas más. Claro, si todo lo hiciera el micro, se perdería velocidad. Por ello es que además de él existen en la máquina otros circuitos integrados que se encargan de "auxiliarlo" y se llaman periféricos. En síntesis, la primera zona controla el computador y guarda el lenguaje que se va a usar en él.

**2:** Al ejecutar una instrucción el micro (como en el BASIC) quizás necesite guardar algún valor o variable. Para ello dispone de la zona de trabajo en donde puede operar con comodidad sin interferir con la zona donde están alojados nuestros programas.

**3:** Cuando ejecutamos GOSUB'S o FOR-NEXT'S, la línea (dirección) de donde partimos es guardada por el micro; en la medida que se acumulen estas direcciones se irán "apilando" en la zona de almacenaje. Este procedimiento permite que siempre la última dirección esté arriba de todo y, por ende, disponible primero; así se puede recuperar y retornar al punto de partida.

**4:** La zona de control es usada para almacenar toda información referida a la programación interna de la máquina, es decir, lo referente al modo de pantalla en uso, las columnas, el color, etcétera; ya que sin ella el microprocesador perdería el control de la máquina y nosotros la cordura.

Si la primera zona se borrara, la computadora quedaría completamente inútil porque el micro no sabría qué hacer; afortunadamente, para evitarlo, se grabó de fábrica en una memoria de lectura solamente (ROM), con lo que se garantiza la perdurabilidad. En cambio, las otras tres, como el micro debe acceder periódicamente para actualizar o colocar datos, deben ser memorias de escritura y de lectura (es una parte de la que usamos para manejarnos nosotros), por ello debemos tener sumo cuidado en no modificar los valores que se encuentran alojados en ellas.

La primera zona comienza en la dirección 0 y tiene 32768 bytes de largo (BYTE se le denomina a la unidad de información, ya sea símbolo, dato, o carácter). Para poder programar en lenguaje de maquina (ASSEMBLER) debemos primero conocer un poco lo que tiene y hace el microprocesador Z-80 que posee nuestra MSX, y así familiarizarnos con nombres y modos de direccionamiento (de programación).

## EXTRATERRESTRE

Las características de nuestro micro son
CAPACIDAD DE DIRECCIONAMIENTO : 2 BYTES
CAPACIDAD DE DATOS : 1 BYTE
ACUMULADORES : 1
REGISTROS NORMALES : 6
REGISTROS INDICES : 2
FLAGS : 1
PUNTEROS : 1
REGISTROS AUXILIARES : 8
PORT'S I/O : 256
PORT'S INP : 65535

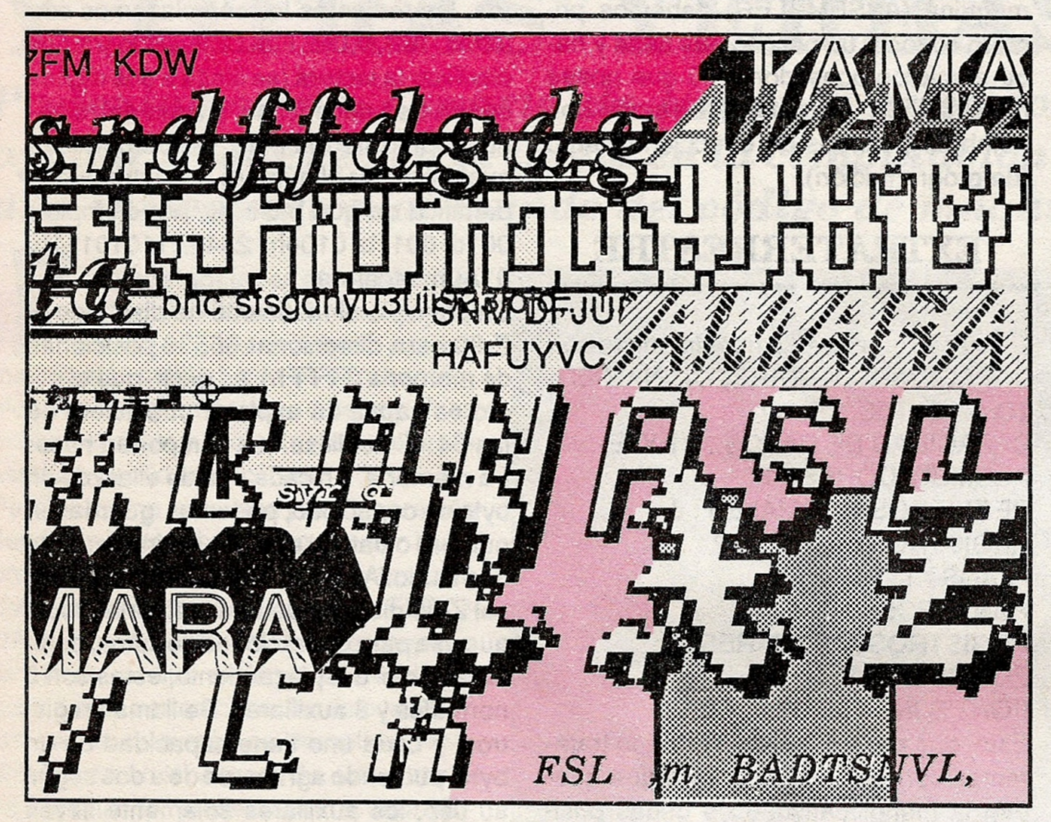
Esto, que parece extraterrestre, lo trataremos con detenimiento para que queden la menor cantidad de dudas posibles. El micro no entiende de números normales (en base decimal), solo comprende dos estados (SI-NO, ALTO-BAJO, "1"-"0"), denominados BIT, que están agrupados de a 4. Por ejemplo, un número o instrucción puede ser SNNS, ABBA o 1001, esto se traduce en un número llamado NIBBLE y tiene 16 combinaciones posibles (de 0000 a 1111). Nuestro microprocesador puede usar dos de estos configurando un BYTE; la cantidad total de combinaciones entonces llega a 256 (de 00000000 a 11111111), pero esta forma de escribir nos dificultaría el seguimiento de un programa en Assembler. Es por ello que se recurre a un equivalente numérico, es decir, que utilizaremos otra base. Nosotros usamos generalmente la base decimal, basada en los números del 0 al 9, pero en lenguaje de máquina es más práctico usar la base hexadecimal (como no tenemos más que diez números, los restantes lo reemplazamos con letras -A,B,C,D,E,F-). Con ello logramos clarificar la lectura (se limita a dos dígitos contra tres del decimal y ocho del binario). Como norma para no perderse se suele asociar al número una letra que identifica de qué base es, por ejemplo : 001d, 0011b, 0100h, 234d, 11010111b, 01A0h, etcétera.

En principio el fabricante ha diseñado el Z-80 para direccionar 65536 posiciones de memoria (FFFFh), es decir, que el micro es capaz de explorar cualquiera de las 64 K celdillas que componen nuestra memoria. En cada una de ellas, de un byte de capacidad, podemos guardar un número o dato, y el micro los inmterpreta uno a uno. Además de tener 64K de RAM , el Z-80 dispone de una memoria propia que usa para operaciones de suma, comparación o desplazamiento, estas son 8 normales y 8 auxiliares. Se llaman registros, y cada uno tiene capacidad de un byte, pudiendo agruparse de a dos según su uso, los auxiliares solamente sirven para guardar datos momentáneamente. Además cuenta con uno llamado acumulador (A) en el que realiza la mayor parte de los cálculos. Para indicar si un resultado es positivo si excede de uno o dos bytes, si es positivo, etcétera, dispone de otro llamado FLAG en el que cada bit indica un estado en particular.

Así como en el BASIC, el micro tambien dispone de una zona de RAM para almacenaje de datos, que va apilando y sacando a medida que lo necesita. Para no perderse se vale de otro registro llamado STACK POINTER (puntero de pila) que constantemente indica dónde está el último elemento ingresado .

# INVASION DE LETRAS

**• CLASE: JUEGO • AUTOR: HORACIO FERRARI**




Este juego, que ha participado del tercer concurso de programas de nuestra revista, nos propone pasar un buen rato tratando de resolver los anagramas que presenta.

Es una adaptación de un viejo juego desarrollado para una computadora TI-99/4A. Como ambas máquinas poseen el mismo chip que controla el video -el TMS-9928- y la TI funciona en el modo que en MSX se llama SCREEN 1, muchos de estos entretenimientos, en los cuales no se requiere un gran movimiento de Sprites, sino pantallas casi fijas (como Escoba de Quince o Truco) son fácilmente transformables con algunas rutinas como las que lleva este programa. Las mismas emulan algunas de las características de la primera máquina. El juego en sí consiste en adivinar en un lapso no mayor de un minuto una palabra que se presenta en pantalla con todas sus letras desordenadas.

Pueden jugar hasta cuatro competidores que la computadora irá designando por orden. Cada palabra es elegida de un banco que se puede ampliar; solamente está limitado por la cantidad de memoria de la computadora.

Las subrutinas fueron puestas al comienzo del programa para lograr mayor velocidad de ejecución y son las que se detallan a continuación.

**1 a 5:** Emulación de CALL COLOR
**8 a 11:** Emulación de CALL CHAR
**14 a 18:** Emulación de CALL HCHAR
**21 a 26:** Emulación de CALL VCHAR
**29 a 34:** Emulación de CALL GCHAR
**1090 a 2270:** Bloque principal del programa.
**2280 a 3000:** Banco de datos de las palabras.

Los parámetros de las subrutinas emuladoras se adaptan tanto en los valores como en el orden de entrada para facilitar la transformación.

El banco de palabras puede ser, como dije antes, ampliado con el solo límite de la memoria de la máquina. Solamente debe respetarse como última palabra la de "FIN", que es usada como limitadora de carga. Asimismo deberá ser ampliado en la línea 1120 el DIM WS$(300) al nuevo valor requerido.

```
1 GOTO 1090:' sigue subrutina de call color
2 VPOKE 8195+CO(1),16*(CO(2)-1)+(CO(3)-1):RETURN
3 'CO(1)=GRUPO CARACTER
4 'CO(2)=COLOR FRENTE
5 'CO(3)=COLOR FONDO
6 '--------------------------------
7 '--------------------------------
8 'SBR.DEFINIC.CARACTERES SCREEN1
9 FOR ZZ=CA(1)*8 TO CA(1)*8+7:VPOKE ZZ,VAL("&H"+MID$(CA$,TI+1,2)):TI=TI+2:NEXT Z
Z:TI=0:RETURN
10 'CA(1)=NUMERO EN ASCII DEL CARACTER
11 'CA$=NUEVA DEFINICION EN HEXAD.
12 '--------------------------------
13 '--------------------------------
14 'SBR.COLOC.HORIZONTAL CARACTERES.
15 LOCATE HC(2)-1,HC(1)-1:PRINT STRING$(HC(4),CHR$(HC(3))):RETURN
16 'HC(1)=POSICION FILA
17 'HC(2)=POSICION COLUMNA
18 'HC(3)=NUMERO EN ASCII DEL CARACT.7060 'HC(4)=CANTIDAD DE REPETICIONES
19 '--------------------------------
20 '--------------------------------
21 'SBR.COLOC.VERTICAL CARACTERES.
22 FOR ZZ=VC(1)-1 TO VC(1)+VC(4)-1:LOCATE VC(2),ZZ:PRINT CHR$(VC(3)):NEXT ZZ:RET
URN
23 'VC(1)=POSICION FILA
24 'VC(2)=POSICION COLUMNA
25 'VC(3)=NUMERO EN ASCII DEL CARACT.
26 'VC(4)=CANTIDAD DE REPETICIONES.
27 '--------------------------------
28 '--------------------------------
29 ' SBR.DETECCION CARACTER EN PANTA-        LLA
30 GC(3)=VPEEK(6144+(32*GC(1)+GC(2)))
31 RETURN
32 'GC(1)=COORDENADA FILA
33 'GC(2)=COORDENADA COLUMNA
34 'GC(3)=CODIGO ASCII CARACTER DETEC        TADO SEGUN LAS COORDENADAS ANTES
  DADA.
35 '--------------------------------
40 ' CONTEO DE TIEMPO
41 TM=TM-1
42 RETURN
43 '--------------------------------
44 '
45 ' IMPRESION CIELO ESTRELLADO
```

```
46 FOR A=1 TO 40:XX=RND(TIME)*30:YY=RND(-TIME)*23:LOCATE XX,YY:PRINT" ":NEXT:RET
URN
47 '-------------------------------
48 '
1090 '  Invasion Alfabetica                    9-3-87
1100 '  Adaptacion de Horacio Ferrari
1101 ON STOP GOSUB 10000:STOP ON
1104 PLAY"O4DECO3CG"
1105 KEY OFF:CLS:SCREEN 1:WIDTH 32:COLOR 15,1,1
1106 DEFINT A-Z:POKE &HFCAB,255:OUT 170,26
1107 DEF FNSO(HZ)=INT(1789800#/(16*HZ))
1115 X=2:Y=-1
1120 DIM WS$(300),NL(15),WA(15)
1130 NW=NW+1
1140 READ WS$(NW):'corresponde a variable WORD$(NW)
1150 IF WS$(NW)<>"FIN" THEN 1130 ELSE NW=NW-1
1160 CA(1)=96:CA$="3C7EFFFF7E3C0000":GOSUB 8
1165 CA(1)=46:CA$="0000000000000100":GOSUB 8
1170 CA(1)=97:CA$="1818181818181818":GOSUB 8
1180 CA(1)=104:CA$="0000000000000000":GOSUB 8
1190 CA(1)=112:CA$="10234719F618C291":GOSUB 8
1200 CA(1)=113:CA$="048365718496F719":GOSUB 8
1210 CA(1)=114:CA$="748D7391F3831174":GOSUB 8
1220 CA(1)=120:CA$="0000FFFFFFFF0000":GOSUB 8
1225 GOSUB 45:'CAMPO ESTELAR
1231 CO(1)=9:CO(2)=9:CO(3)=1:GOSUB 2
1232 CO(1)=10:CO(2)=1:CO(3)=1:GOSUB 2
1233 CO(1)=11:CO(2)=11:CO(3)=1:GOSUB 2
1235 CO(1)=12:CO(2)=14:CO(3)=1:GOSUB 2
1240 FOR L=2 TO 8
1250 CO(1)=L:CO(2)=16:CO(3)=1:GOSUB 2
1260 NEXT L
1270 LOCATE 7+X,15+Y:PRINT"I N V A S I O N":LOCATE 5+X,21+Y:PRINT"D E    L E T
R A S"
1280 FOR L=1 TO 3000:NEXT L
1290 SCREEN,1
1300 SPRITE$(1)=CHR$(&H3C)+CHR$(&H7E)+CHR$(&HFF)+CHR$(&HFF)+CHR$(&H7E)+CHR$(&H3C
)
1310 FOR L=2 TO 115
1320 PUT SPRITE 1,(L,20),5,1:'CALL SOUND(-1,500-L,7-3,0)
1325 SOUND 8,10:SOUND 0,FNSO(500-L) MOD 256:SOUND 1,INT(FNSO(500-L)/256)
1330 NEXT L
1340 VC(1)=4:VC(2)=15:VC(3)=97:VC(4)=11:GOSUB 21:'CALL SOUND(20,1500,5,-6,0)
1345 SOUND 8,15:SOUND 0,FNSO(1500) MOD 256:SOUND 1,INT(FNSO(1500)/256):FOR A=1 T
O 10:NEXT:SOUND 8,0
1350 VC(1)=4:VC(2)=15:VC(3)=104:VC(4)=11:GOSUB 21
1360 'CALL SOUND(150,110,9,-6,0)
1365 SOUND 8,15:SOUND 0,FNSO(110) MOD 256:SOUND 1,INT(FNSO(110)/256):FOR A=1 TO
15:NEXT:SOUND 8,0
1370 FOR L=1 TO 4
1380 FOR L2=112 TO 114
1390 HC(1)=15:HC(2)=8:HC(3)=L2:HC(4)=16:GOSUB 14
1400 NEXT L2:NEXT L
1410 HC(1)=15:HC(2)=7:HC(3)=104:HC(4)=20:GOSUB 14
1420 VC(1)=4:VC(2)=15:VC(3)=97:VC(4)=17:GOSUB 21:'CALL SOUND(20,1500,5-6,0)
1425 SOUND 8,15:SOUND 0,FNSO(1500) MOD 256:SOUND 1,INT(FNSO(1500)/256):FOR A=1 T
O 10:NEXT:SOUND 8,0
1430 VC(1)=4:VC(2)=15:VC(3)=104:VC(4)=17:GOSUB 21
1440 'CALL SOUND(150,110,9,-6,0)
1445 SOUND 8,15:SOUND 0,FNSO(110) MOD 256:SOUND 1,INT(FNSO(110)/256):FOR A=1 TO
15:NEXT:SOUND 8,0
1450 FOR L=1 TO 4
1460 FOR L2=112 TO 114
1470 HC(1)=21:HC(2)=7:HC(3)=L2:HC(4)=21:GOSUB 14
1480 NEXT L2:NEXT L
1490 HC(1)=21:HC(2)=7:HC(3)=104:HC(4)=21:GOSUB 14
1500 FOR L=116 TO 255
1510 PUT SPRITE 1,(L,20),5,1:'CALL SOUND(-1,380+L,7,-3,0)
1515 SOUND 8,10:SOUND 0,FNSO(380+L) MOD 256:SOUND 1,INT(FNSO(380+L)/256)
1520 NEXT L:PUT SPRITE 1,(L,208)
```

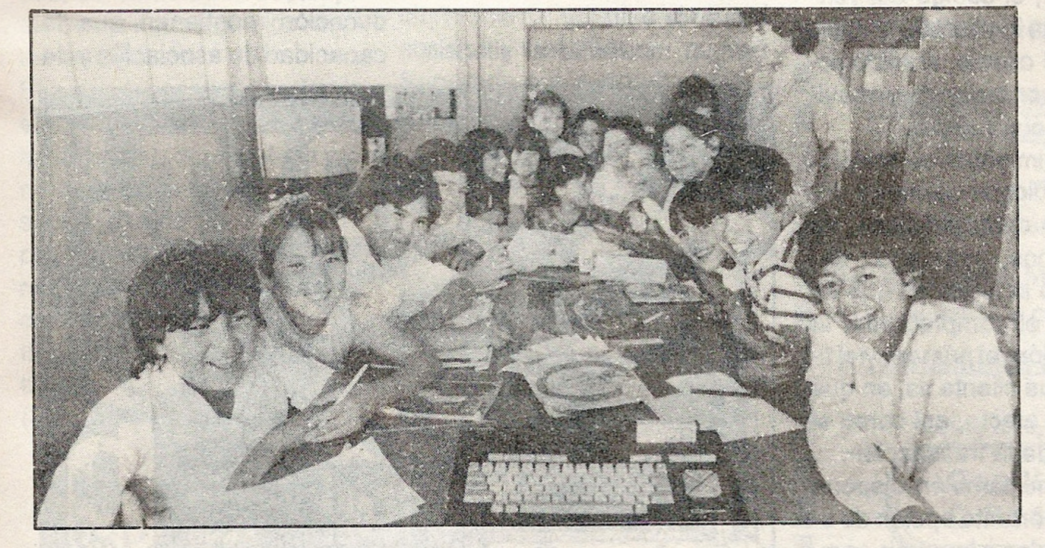
```
1530 CLS
1540 BEEP:LOCATE 1+X,4+Y:PRINT"NUMERO DE JUGADORES? (1-4)"
1545 A$=INKEY$:IF A$=""THEN 1545 ELSE IF VAL(A$)=1 OR VAL(A$)=2 OR VAL(A$)=3 OR
VAL(A$)=4 THEN 1550 ELSE 1540
1550 NP=VAL(A$):FOR L=1 TO NP
1560 LOCATE 1+X,6+(L*2)+Y:PRINT "NOMBRE JUGADOR";L:LOCATE 20+X,6+(L*2)+Y:BEEP:IN
PUT PN$(L)
1565 IF LEN(PN$(L))>9 THEN PN$(L)=LEFT$(PN$(L),9)
1570 NEXT L
1580 CLS
1585 GOSUB 45
1590 HC(1)=16:HC(2)=1:HC(3)=120:HC(4)=32:GOSUB 14:HC(1)=19:HC(2)=1:HC(3)=120:HC(
4)=32:GOSUB 14
1600 LOCATE 1+X,18+Y:PRINT"TIEMPO":LOCATE 12+X,18+Y:PRINT"PALABRA: "
1610 LOCATE 7+X,20+Y:PRINT"** PUNTAJE **"
1620 LOCATE 1+X,22+Y:PRINT PN$(1);:LOCATE 16+X,22+Y:PRINT PN$(2);
1630 LOCATE 1+X,23+Y:PRINT PN$(3);:LOCATE 16+X,23+Y:PRINT PN$(4);
1640 FOR PL =1 TO 5
1650 FOR L=1 TO NP
1660 LOCATE 0,1+Y:PRINTSTRING$(32," "):LOCATE 8+X,1+Y:PRINT"TURNO: ";PN$(L)
1670 GOSUB 1700:GOSUB 1810:GOSUB 1970
1680 NEXT L
1690 NEXT PL:GOTO 2210
1700 WO=1+INT(RND(-TIME)*NW):IF WS$(WO)="" THEN 1700
1710 WO$=WS$(WO)
1720 WS$(WO)=""
1730 NL=LEN(WO$):SC$="":IF NL>10 THEN 1700
1740 FOR L2=1 TO NL:WA(L2)=0:NEXT L2
1750 FOR L2=1 TO NL
1760 N=1+INT(RND(-TIME)*NL):IF WA(N)=1 THEN 1760
1770 SC$=SC$+MID$(WO$,N,1):WA(N)=1
1780 NEXT L2
1790 IF SC$=WO$ THEN 1730
1800 RETURN
1810 '
1820 FOR L2=2 TO 84
1830 PUT SPRITE 1,(L2,20):'CALL SOUND(-1,500,7,-3,0)
1835 SOUND 8,10:SOUND 0,FNSO(500) MOD 256:SOUND 1,INT(FNSO(500)/256)
1840 NEXT L2
1850 FOR L2=1 TO NL
1860 VC(1)=4:VC(2)=(L2*2)+9:VC(3)=97:VC(4)=9:GOSUB 21:' CALL SOUND(20,1500,5,-6,
0)
1863 SOUND 8,15:SOUND 0,FNSO(1500) MOD 256:SOUND 1,INT(FNSO(1500)/256):SOUND 8,0
1865 VC(1)=4:VC(2)=9+(L2*2):VC(3)=104:VC(4)=9:GOSUB 21
1870 HC(1)=14:HC(2)=10+(L2*2):HC(3)=ASC(MID$(SC$,L2,1)):HC(4)=1:GOSUB 14
1880 FOR L3=1 TO 16
1890 PUT SPRITE 1,(68+(L2*16)+L3,20),5,1:'CALL SOUND(-1,380,7,-3,0)
1893 SOUND 8,10:SOUND 0,FNSO(380) MOD 256:SOUND 1,INT(FNSO(380)/256):SOUND 8,0
1895 DC=68+(L2*16)+L3
1900 NEXT L3:NEXT L2
1910 '
1920 FOR L2=DC TO 255
1930 PUT SPRITE 1,(L2,20),5,1:'CALL SOUND9-1,500,7,-3,0)
1935 SOUND 8,10:SOUND 0,FNSO(500) MOD 256:SOUND 1,INT(FNSO(500)/256)
1940 NEXT L2:SOUND 8,0
1950 PUT SPRITE 1,(20,208)
1960 RETURN
1970 TI=0:TM=60:AN$="":TIME=0
1980 LOCATE 20+X,18+Y:PRINT " "
1990 K$=INKEY$
2000 IF TM=0 THEN 2050
2005 IF TIME=>50 THEN TM=TM-1:TIME=0
2006 LOCATE 7+X,18+Y:PRINT TM;:IF K$="" THEN 1990
2010 'TI=TI+1:TM=100-INT(TI/30):LOCATE 7+X,18+Y:PRINT TM::IF K$="" THEN 1990
2020 K=ASC(K$):IF K=13 THEN 2070
2030 IF K=8 THEN 2190
2040 AN$=AN$+K$:LOCATE 20+X,18+Y:PRINT AN$;:GOTO 1990
2050 'CALL SOUND (1000,800,5,-6,0)
2055 PLAY"O5C"
2060 LOCATE 6+X,12+Y:PRINT"** FUERA DE TIEMPO **":GOTO 2090
2070 IF AN$=WO$ THEN LOCATE 8+X,12+Y:PRINT "** CORRECTO **" ELSE LOCATE 8+X,12+Y
:PRINT "** ERROR **":TM=0:PLAY "O5A":GOTO 2090
2080 'CALL SOUND(1000,200,4,2000,0)
2083 PLAY "O3A","O5A"
2090 FOR L3=1 TO 8:FOR L4=112 TO 114
2100 HC(1)=14:HC(2)=12:HC(3)=L4:HC(4)=LEN(WO$)*2-1:GOSUB 14
2110 NEXT L4:NEXT L3
2120 HC(1)=14:HC(2)=1:HC(3)=32:HC(4)=32:GOSUB 14:HC(1)=15:HC(2)=1:HC(3)=32:HC(4)
=32:GOSUB 14:LOCATE 11+X,14+Y:PRINT WO$:FOR DE=1 TO 3000:NEXT DE
2130 HC(1)=12:HC(2)=1:HC(3)=32:HC(4)=32:GOSUB 14:HC(1)=14:HC(2)=1:HC(3)=32:HC(4)
=32:GOSUB 14:HC(1)=15:HC(2)=1:HC(3)=32:HC(4)=32:GOSUB 14
2140 LOCATE 8+X,18+Y:PRINT"60";:LOCATE 19+X,18+Y:PRINT "             ";
2150 SC(L)=SC(L)+TM*2
2160 R=22+INT(L/3):C=10:IF L=2 OR L=4 THEN C=C+15
2170 LOCATE C+X,R+Y:PRINT USING"###";SC(L)
2175 INTERVAL OFF
2180 RETURN
2190 IF LEN(AN$)<2 THEN AN$="" ELSE AN$=MID$(AN$,1,LEN(AN$)-1)
2200 LOCATE 20+X,18+Y:PRINT AN$+" ":GOTO 1990
2210 HC(1)=1:HC(2)=1:HC(3)=32:HC(4)=32:GOSUB 14
2220 LOCATE 6+X,1+Y:PRINT "** FIN DEL JUEGO **"
2225 PLAY"O4DECO3CG"
2265 LOCATE 3,8:PRINT"OTRO JUEGO? [ S/N ]"
2266 A$=INKEY$:IF A$=""THEN 2266
2267 IF A$="S" OR A$="s" THEN RUN ELSE IF A$="N" OR A$="n" THEN CLS:PRINT"GRACIA
S POR TU COMPANIA":END
2270 GOTO 2266
2275 '------------------------------
2280 DATA TREN,LIBROS ESCRITORIO,PAPEL,MEDIA,CAMISA,ESCRIBIR,SACON
2290 DATA CALCULA,RESTA,PIENSA,NUMERICO,SUMAR,PANTALLA,MONITOR,IMPRESORA,DIVIDE
2300 DATA REGLA,LAPIZ,ANIMAL,GOMA,VALIJA,CABLE,CORTINA,TORRE
2310 DATA SALSA,INVASOR,PICKLE,CURVA,CALLE,AVENIDA,PLANTA,GRAFICO,NOCHE,PERSONA,
RADICAL
2320 DATA GIGANTE,LANA,MEDIO,GLOBO,CUADRADO,JINETE,CARNE,AUTOR,VERANO,INVIERNO,M
ETODO
2330 DATA PISTOLA,ROJO,ROTUNDO,REDONDO,SONRISA,TRANSITO,TRAMPA,VALIENTE,PESO,PES
ADO,MUJER
2340 DATA ZAPATO,ZANAHORIA,PROYECTO,JUNGLA,CABEZA,GRAN,JUEZ,RAPIDO,DURO,VERDAD,B
IZCOCHO,ARMONIA,MUSICAL,ESCALA
2350 DATA RADIO,RELOJ,VALOR,VECTOR,CHICO,MESES,PERRO,PAJARO,PROGRAMA,CADENA,GATO
,TRAZO
2360 DATA DIRECCION,PUNTO,LOGICA,VERSION,ENANO,COMPANIA,RAZON,GRADO,MUNDO,SILLA,
GUIA,REINA,TORMENTA
2370 DATA MAGIA,RESOLVER,MAQUINA,MAESTRO,CAPITAL,OSCURO,RECORDAR,EDUCAR,PRODUCE,
ELIJE
2380 DATA DERECHO,ENTRA,HELADO,ASPIRINA,BANANA,AFUERA,NORMAL,MAYOR,ACENTO,MEJOR,
ANANA,ESPIRAL,REMOTO
2390 DATA SILLON,CARRERA,PLAYA,LISTAS,JUNTO,CAVIDAD,HUESO,MANUAL,ECONOMIA,EXPAND
IR,PERAS,DIESTRO,REPARAR
2400 DATA PRESO,SONIDO,DISCOLO,TRETA,IMPULSO,IMAGINAR,PETARDO,LLEGAR,LAMENTO,EVI
TAR,TRAER,FUTURO,AVERIA,JUSTICIA
2410 DATA TELEFONO,CALLE,VENTILADOR,SIFON,ALMANAQUE,PLAYAS,MUNDO,SALAME,CABEZA,P
APA
2420 DATA MATE,SOL,LUNA,ESTRELLAS,MINERVA,POMO,PERFUME,PINCEL,PINTURA,COLAS,ANTE
NA,CARAMELO,GOLOSINA,GRILLO,CARTA
2430 DATA CRISTAL,AVION,BARCO,SUBMARINO,REVISTA,TINA,BILLETERA,TECLA,ARBOL,FLOR,
VIENTO,HUMEDAD,TEMPERATURA,HORA
2440 DATA DIBUJO,PILA,GOTA,NUBES,AIRE,MAR,PLATA,OLAS,CENTRO,POSTRE,HUESOS,ARMONI
OSO,HUESPED
2450 DATA DISCO,VEREDA,NENA,MOLINO,PILETA,TELEFONO,LLAVES,ZOCALOS,LLUVIA,MORTADE
LA,CARPETA,ENCHUFE,CIELO,NAVES,HOJAS,LENTES
2460 DATA ABANICO,ABEJAS,BALANZA,BARRILETE,DISFRAZ,ENREDADERA,ESPIGA,MONEDA,MERC
ADO,MARINERO,TENEDOR,TOALLA,TOLDO,TINTA,SEMANA,SERRUCHO,SAUCE,TITERES
2470 DATA TRINEO,TREBOL,TRANVIA,URRACA,UVA,TROMPO,VOLCAN,YACARE,YUYO,ZAPALLO
2480 DATA PRIMAVERA,CAPILLA,MONTE,ROSA,SIERRA,RIO,CORCHO,PATIO,OMNIBUS,TRANSPORT
E,CANAL,TORRE,TIEMPO,VACACIONES,HISTORIETA,HIJO,AUTOMOVIL,CAMION,MOTO,MOTONETA
2490 DATA VIDEO,RECIBO,PLANETA,TECLA,GRABADOR
3000 DATA FIN
9999 END
10000 SCREEN 0:COLOR 15,4,4:KEY ON:'SBR reinicializacion despues de STOP
```

# EXPERIENCIA EN LA ISLA MACIEL

***Através de un esfuerzo conjunto, un grupo de niños cobijados en un centro comunitario de un barrio popular tiene en la computación un nuevo elemento para su evolución.***



En la Isla Maciel (provincia de Buenos Aires) funciona el Centro Comunitario Nº 5 "Gral San Martín", donde se está desarrollando, desde hace cuatro meses, una experiencia de informática con niños de 7 a 13 años.
La misma es posible gracias al apoyo de Telemática y de Editorial PROEDI, a través de sus revistas K64 y LOAD MSX, que aportaron una computadora, material bibliográfico y software.
Dirige el centro la Hermana María Dolores y los cursos son dados por el señor Humberto González. Asisten a ellos unos treinta chicos, que están divididos en tres niveles.
En el primero, con alumnos de hasta tercer grado, se enseña a reconocer las letras y los números. Los niños aprenden también a hacer caminar y girar a la tortuga, usando su propio cuerpo como referencia. Esto está basado en las teorías de Papert.
En el segundo nivel (cuarto y quinto grado), se comienza a identificar a las figuras geométricas y sus elementos. Esto se logra asociando las diferentes figuras con objetos conocidos. Por ejemplo, el cuadrado se asocia a una manzana, siendo los lados las diferentes cuadras y los ángulos, las esquinas.
Finalmente, los más grandes (sexto y séptimo grado) utilizan las figuras elementales para hacer otras más complicadas. También se ve diagramación e instrucciones en BASIC.
Los alumnos realizan primero el trabajo en cuadernos, y luego lo pasan a la computadora.
Los proyectos para el año venidero son estructurar los cursos, sobre la base de la experiencia de este año e ingresar poco a poco en el mundo de la cibernética.
El proyecto consiste en aprender cómo trabaja el cuerpo, cuáles son los movimientos y qué función cumple cada órgano. La idea es contar con un pequeño robot para ir mostrando qué parte está funcionando para determinada tarea.
Como nos dijo González el objetivo es que los chicos razonen, y que puedan usar la computadora como un medio para ello.
Editorial Proedi colabora así con este centro, de manera tal que los chicos aprovechen su tiempo libre para aprender y para jugar. el objetivo es sacar a estos niños de la calle, para introducirlos en un ámbito positivo.
Quienes quieran ponerse en contacto con ellos, para enviarles programas, libros y otros elementos de computación, pueden dirigirse a:
Centro Comunitario Nº 5 "General San Martín"
**Pasaje Los Aromos**
**Barrio San Martín**
**(1871) Isla Maciel**
**Provincia de Buenos Aires**
**Tel: 201-4040**



## CUENTAS CORRIENTES

```
EMPRESA
DIRECCION
()
T.E:TELEFONO
Nro. de Ganancias :
Nro. Ingr. Brutos :
Nro. I.V.A.        :IVA
Impuestos Internos:

                                                ========================
                                                !   F A C T U R A      !
                                                ------------------------
                                                !          1           !
                                                ========================
    Sres:
        FERNANDEZ, Luis Alberto
        Ayacucho 1356  5:P
        (1426)Capital
        Nro. GANANCIAS:
        TIPO DE I.V.A.:GENERAL
        Nro. I.V.A.   :234543
        Nro.Cta.: 98          , 15 de Diciembre de 1987
```

| CANT. | C.ART. | DESCRIPCION | P.UNIT. | IMPORTE |
|---|---|---|---|---|
| 2 | 22 | computadoras | 540.00 | 1080.00 |
| 4 | 15 | calculadoras | 80.00 | 320.00 |
| 8 | 45 | relojes | 64.50 | 516.00 |
| 1 | 83 | impresoras | 745.00 | 745.00 |
| | | SUBTOTAL | A | 2661.00 |
| | | IVA 18% | A | 478.98 |
| | | TOTAL GENERAL | A | 3139.98 |

SON AUSTRALES :TPES MIL CIENTO TREINTA Y NUEVE CON 98 CTVS

La empresa COMPUTAR S.R.L. ha lanzado al mercado un nuevo sistema de cuentas corrientes, para ser operado con computadoras de la norma MSX.

Este sistema permite manejar todas las actividades vinculadas al control de cuentas corrientes. Por ejemplo, listados económicos, financieros, impositivos y de revisión de movimientos, impresión de comprobantes, etcétera.

La operación del sistema no requiere conocimientos de computación, sino que todos pueden utilizarlo. El único requerimiento es familiarizarse con el manejo de las diferentes pantallas.

El equipo contiene:
- 1 cartucho ROM
- 1 disquete con el sistema de cuentas corrientes.
- 1 disquete con el manual de operación.

El sistema está orientado al control de las cuentas corrientes en un comercio o pequeña industria.

Está compuesto por una serie de programas que permiten ingresar, visualizar, imprimir y eliminar información de un grupo de archivos. Los principales son el de clientes, donde se almacenan los datos de los mismos, y el de movimiento, que detalla las variantes en el estado de las cuentas corrientes.

Cada archivo de clientes permite guardar para un determinado número de personas los siguientes ítems: nombre, dirección, localidad, código postal, código de provincia, teléfono, número y tipo de I.V.A., el código del vendedor que lo atiende, la categoría del cliente, el saldo actual, observaciones y movimientos que realizó.

Los movimientos se relacionan con dichos clientes y detallan las características de los mismos, guardando datos como la fecha en que se produce el comprobante, la descripción, el número y el tipo, de qué cliente es, en qué cantidad afecta, así como el importe de la transacción.

El sistema también dispone de la importante opción de emitir listados e imprimir comprobantes, y permite de esta manera volcar al papel los distintos tipos de datos.

(COMPUTAR)

## COLT 36

Las sonoras pisadas de las botas con espuelas retumbaron junto a la puerta del " saloon" y se detuvieron allí, esperando. Adentro, la fiesta estaba en su apogeo: grupos de vaqueros se amontonaban en las mesas, bebiendo o jugando al póker. Solo la barra estaba relativamente vacía, ocupada por cuatro hombres de torva expresión. Hombres altos con pistoleras dobles y manos enguantadas en cuero.

Entonces, poco a poco, más y más ojos captaron la presencia de la figura alta que se recostaba en la puerta con la insignia de la estrella prendida en el pecho. Casi al momento se inició un desbande general. En instantes el local estaba vacío, excepto por los cuatro hombres de la barra, vueltos ahora hacia la puerta y con las manos muy cerca de las cananas.

- Creíamos que se habían ocupado de ti en el cañón- dijo uno de ellos.

El sheriff traspuso el umbral.

- Ellos también lo creyeron- respondió

- ¿Qué quieres ahora?

- A ustedes - contestó el hombre de la estrella.

Por un momento el silencio reinó en el lugar, mientras los cinco hombres se enfrentaban con la mirada. Luego, el jefe de los pistoleros dio una señal casi invisible y cinco manos volaron hacia las fundas de los revólveres.

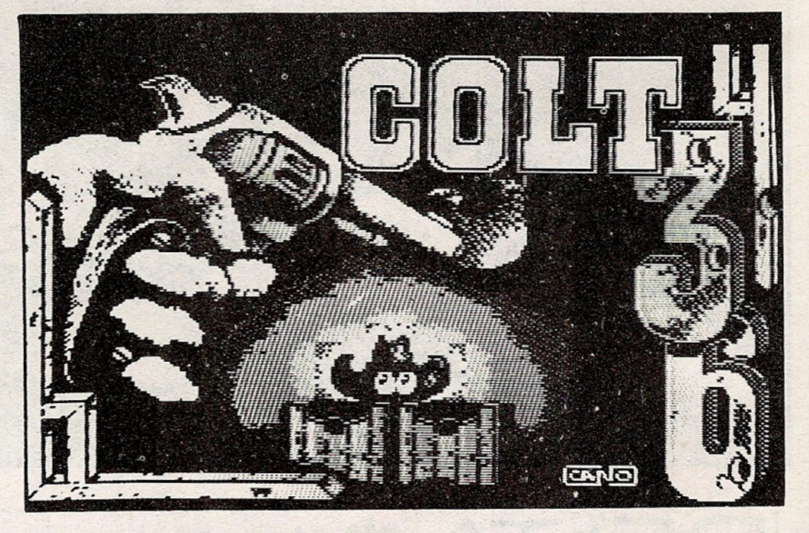
El juego consiste en acabar con todas las peligrosas bandas de pistoleros, rebeldes, indios, etcétera, repartidos por diversos escenarios: un almacén, un cañón, una mina y un "saloon", con nivel creciente de dificultad. Además de disparar contra bandidos, podemos romper botellas, cristales de ventanas, apagar velas, abatir pájaros, caballos, etcétera.

Disponemos de un punto de mira que podemos mover sobre un amplio campo de visión. Para ayudarnos, contamos con un individuo que nos indica por dónde van a salir los pistoleros. (VALENTE)

## VIAJANDO CON HELI 1

El objetivo de este juego es que el niño desarrolle la capacidad de establecer relaciones de asociación entre objetos dados, en función de su uso, proximidad, forma, color, cantidad, etcétera.

Los autores del entretenimiento, profesionales de la educación, sostienen que la capacidad de asociación y de




simbolización dotan al niño para el aprendizaje de la lecto-escritura y el cálculo. Sin estas capacidades no sería posible alcanzar ni una buena mecánica de lectura (por ejemplo, discriminar un sonido, cualquiera sea su posición, y asociarlo al signo que lo representa), ni una sólida comprensión lectora, ni tampoco la abstracción numérica (de la cantidad y su asociación al signo que la representa).

Los elementos del juego se

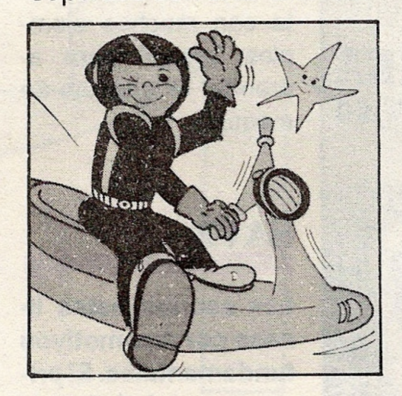
reúnen en varios grupos: animales, deportes, personajes. Dichos elementos aparecen tres en cada juego para ser asociados a otros juegos diferentes.
En la pantalla aparece en la zona superior, un pajarito que vuela a su nido, y que simboliza el concepto que se trabaja.
En la zona inferior el movi-




miento ascendente del Sol o de la Luna representa los aciertos logrados. En la zona central o zona de juego, aparecen tres imágenes de alguno de los tres grupos ya citados, así como otros tres objetos con los que aquellos mantienen relación.
El juego se realiza mediante el correcto manejo de HELI (el helicóptero), que traslada y une los objetos que hay que asociar. Cuando el niño completa con éxito una de las series presentadas, y antes de comenzar con la siguiente, tiene acceso al juego recreativo.
(SYSTEMAC)

## LA MOTO ESPACIAL 2

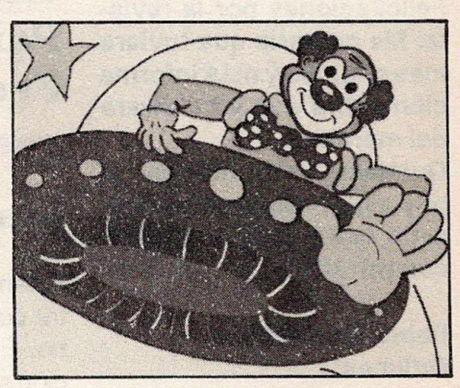
Este entretenimiento es otro más de la serie Aprender Jugando de SYSTEMAC.
El objetivo del mismo es desarrollar en el niño la memoria visual de los objetos respecto a su forma y a su posición. Al mantenerse la imagen durante breves instantes, el niño agiliza y desarrolla su capacidad de memoria visual por tener que recordar con posterioridad los objetos presentados, así como el lugar que ocupaban en la pantalla.
En este juego se le muestran gradualmente al niño uno, dos o tres objetos durante unos instantes para que los memorice y, a continuación, los reconozca entre otros.
Hay tres niveles de dificultad. En el nivel uno el niño debe seleccionar de entre los cinco objetos aquel que sea igual al que ha centelleado previamente.
Luego lo tomará y depositará, mediante la MOTO ESPACIAL, en el recuadro que aparece en la pantalla.
En los siguientes niveles, se debe recordar no sólo los dos o tres objetos que han parpadeado, sino también la posición que ocupaban.
Si en el desarrollo del juego el niño comete un error de selección o posición, volverán a parpadear los objetos de los recuadros no rellenados, y continuará el juego.
La información visual de acierto o error se refuerza con sonidos y melodías.
Superada con éxito una fase



del entretenimiento, y antes de comenzar con la siguiente, el chico puede tener un recreo o con un pasatiempo que le propone el programa.
(SYSTEMAC)

## EL CAZADOR DEL ESPACIO 2

Este programa cuenta con dos juegos. En el primero el objetivo es que el niño forme parejas de objetos diferentes, pero de la misma altura, a partir de la comparación.
En el segundo entretenimiento el objetivo es emparejar objetos de la misma longitud, independientemente de la forma y/o color.
Al alcanzar este objetivo, el niño abstrae la noción de tamaño como cualidad de los objetos al margen de su color, forma o posición, y se prepara para realizar clasificaciones de objetos en función de una cualidad: el tamaño.
El niño desarrolla de esta manera su destreza manual y su sentido espacial al tener que mover una nave para posarla sobre el objeto elegido y transportarlo por una pantalla hasta el lugar adecuado.
Los juegos se componen de 16 elementos diferentes; en uno se busca la diferencia entre grandes y pequeños y en el otro entre largos y cortos.
Tanto en un entretenimiento como en el otro, el robot que aparece en la parte inferior izquierda crecerá con los aciertos logrados y agitará los bra-

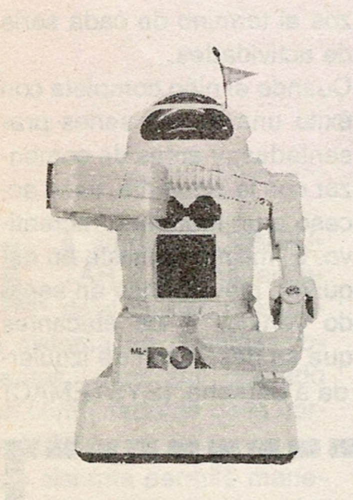


zos al término de cada serie de actividades.
Cuando el niño completa con éxito una de las series presentadas, y antes de comenzar con la siguiente, tiene acceso a un momento recreativo. El mismo consiste en esquivar, moviéndose en sentido vertical, a los atacantes que se desplazan de izquierda a derecha. (SYSTEMAC)

## ROBOTS

**Felicitaciones por la revista. Me gustaría que tuviera más páginas y más información sobre soft y hardware del mercado.**
**Quisiera preguntarles si hay algún robot en el mercado que pueda comandar con mi MSX.**

**Pablo Battistoni**
**Quilmes**



## Load MSX

Dada la extensión de esta sección no podemos darte todos los detalles aquí. Próximamente publicaremos una nota sobre el tema.
Te podemos ir anticipando que sí existen robots en el país, tanto de fabricación nacional como importados. Por ejemplo están el RBS-1 diseñado por R. Tokuda, H. Caro y E. Graffigno y el ROBOT ROBO MSX.

## OPINION

**Con respecto a vuestro loable pedido de comentarios sobre la revista, considero que sería conveniente aumentar la cantidad de software, en lo que a utilitarios se refiere, así como mayor información sobre novedades del exterior.**

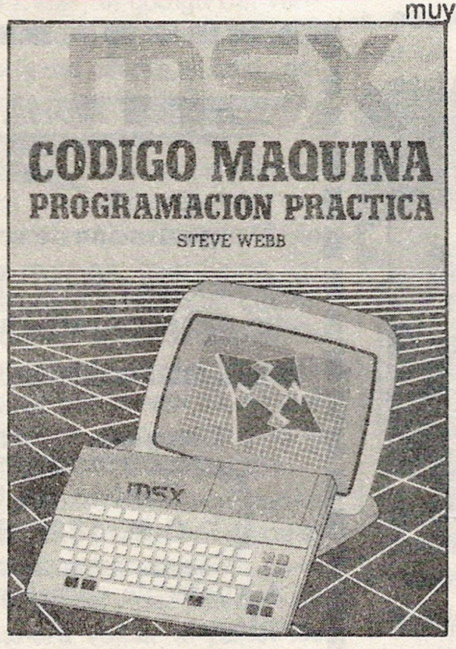
**Eduardo Accialini**
**Escobar**

**Para comunicarse con nosotros deben escribirnos a "Load Revista para usuarios de MSX", Paraná 720, 5º Piso, (1017), Capital Federal.**

## Load MSX

Estamos tratando de comentar cada vez más soft utilitario. En cuanto a la información sobre el exterior también tomaremos en cuenta tu pedido. Prueba de ello es la nota sobre el MSX-FAX, publicada en este número.
Te agradecemos tus opiniones y reiteramos la invitación a nuestros lectores para que nos envíen comentarios sobre nuestra revista.

## LIBROS




**Antes que nada quiero felicitarlos por la excelente publicación.**
**Estoy suscripto desde abril y no me arrepiento pues me presta gran ayuda.**
**Me gustaría que me recomendaran un libro para aprender lenguaje de máquina, ya que me gustaría estar informado al respecto.**

**Aldo Oscar Chernutti**
**Mar de Ajó**

## Load MSX

Nos alegramos de poder brindarte ayuda mes a mes. Hay muchos textos que tocan el tema del lenguaje de máquina en el Z80 y cualquiera de ellos puede ser de mucha ayuda.
Por ejemplo, te podemos recomendar el libro de Steve Webb: **MSX - Código máquina - Programación Práctica.**
Este libro es una muy buena introducción al tema y cuenta con muchos ejemplos "piolas" para aprender el lenguaje de máquina.

## SAVE

**Les escribo estas líneas por dos motivos fundamentales. El primero es felicitarlos por la revista. No hay duda que es la mejor revista del país dedicada a la norma MSX. Es cierto que no hay otra, pero si la hubiera, sería difícil que alcanzara su nivel.**
**El otro motivo por el cual les escribo es una consulta con respecto a la grabación de archivos en código ASCII. ¿Con ello podría pasar los listados a un procesador de texto y allí darle el formato que yo quiera?**
**Sigan adelante.**

**Marcela Benet**
**Capital Federal**

## Load MSX

Agradecemos tus conceptos y pasamos a contestar tu pregunta. Lo que decís es posible grabando en disco de la siguiente manera: SAVE "nombre del programa",A.
De esta manera tendrás almacenado el programa en código ASCII y luego listarlo, como deseas, en un procesador de texto (por ejemplo el MSX-WRITE)

## SOFT EDUCATIVO




**Felicitaciones por la revista. Compré una Talent porque la necesitaba para mi profesión, soy contador, pero quiero que la misma también les pueda servir a mis dos hijos, Lucas y Juan Pablo, de 12 y 8 años respectivamente. ¿Qué software educativo podría conseguir?**

**Ezequiel M. Palomba**
**Mar Del Plata**

## Load MSX

Le podemos recomendar el software elaborado por SYSTEMAC, que se compone de varias series, para cubrir las diferentes necesidades. Por ejemplo está la serie sobre biología, sobre física (Patágoras), sobre lógica (Logicolor), etcétera. Estos programas, desarrollados por profesionales de la educación plantean al alumno una nueva manera de aprender.

# PROGRAMAS EDUCATIVOS

**Bajo Licencia de: IDEALOGIC , Fisher Price, SM, Alea, Spinnaker y Dimension News**




*** Serie Lexa**
1 El Duende
2 El Tesoro
3 El Torreon
4 El Oasis

*** Serie Logicolor**
1 Autos Locos
2 Manzanas y Gusanos
3 Rehenes

*** Serie Adolescentes**
Invierta y Gane
Compra y Vende
Roma: La Conquista

*** Serie Aprender Jugando**
1 Aventuras en el Circo I
2 Aventuras en el Circo II
3 La Alfombra Mágica I
4 La Alfombra Mágica II
5 Viajando con Heli I
6 Viajando con Heli II
7 El Cazador del Espacio I
8 El Cazador del Espacio II
9 La Abejita Inquieta I
10 La Abejita Inquieta II
11 La Abejita Inquieta II

Idea Base
Idea Tutor
Idea Texto



Edita y Distribuye SYSTEMAC S.A. ESMERALDA 320. 5º PISO, Capital Federal.
Teléfonos 35-1790/7942/1703/6179/9799.

# Encienda una computadora Talent MSX y sus periféricos.

### MODEM
- Interfaz asincrónica incorporada.
- Normas BELL y CCITT, 300 y 1.200 baudios.
- Incluye en 80 Kb software de comunicaciones, MSX-PLAN y MSX-WRITE.

### UNIDAD DE DISCOS FLEXIBLES
- 5 1/4'' de 360 Kb (DS-DD)
- Velocidad transferencia 250 Kb/sg.
- Formato grabacion compatible MS-DOS.

### MINI-LAN
EQUIPO CENTRAL PROFESOR — IMPRESORA
EQUIPO ALUMNO 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 — EQUIPO ALUMNO
DISKETTERAS 2 x 360 KB

### (RED PARA USO EDUCACIONAL)
- Comunicación por línea compartida a 30.000 baudios.
- Comparte unidades de disco e impresoras de consola MSX maestro con hasta 10 consolas MSX alumno
- Carga simultánea de un programa a todos los alumnos.
- Carga individual de maestro a alumno
- Salvado de programa alumno a unidad de disco maestro.
- Salida a impresora de maestro del listado de programa alumno, en spooling.
- Funciona desde MSX-Basic, MSX-Logo o cualquier programa que corra desde Basic.
- Estando activa, se dispone de todo el MSX-Basic.

### RS-232
- Velocidades programables desde 50 a 19.200 baudios.
- Procesamiento de comunicaciones desde Basic.
- Buffer de 128 caracteres para recepción

### TECLADO NUMERICO
- Conexion a Joystick
- Se integra a todo el software que corre bajo MSX-DOS (Ej.: D-Base II, MSX-Plan, etc.)

### MOUSE
- Código DPM-220, accesorio para graficar.

### MONITOR MONOCROMATICO 12''
- Anti-reflejo - Fósforo verde.
- Apto para uso profesional.
- (80 caracteres x 25 líneas).
- Parlante con amplificador incorporado.

### EXPANSION 80 COLUMNAS
- Hace posible la utilización de software estandar CP/M, emulando terminal tipo VT-52.
- Incluye software para manejo de video.

## Software

| MSX-LOGO | MSX-LPC | MSX-PLAN | MSX-WRITE |
|---|---|---|---|
| Desarrollado por Logo Computer System Inc. con aplicación de primitivas y redacción del Manual por los Ings. Hilario Fernández Long y Horacio Reggini. | Lenguaje de programación estructurado y en castellano. | Planilla de calculo de Microsoft Corp. (Versión para MSX del Multiplan.) | Procesador de palabra de ASCII Corp. en castellano. |

